Cyberscope

Audit Report

Fruitcoins

October 2023

Network BSC

Address 0xbA18643Bd61d6030182D11C1eEEf9B0a51c853d4

Audited by © cyberscope

Analysis

Critical Medium Minor / Informative Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

Critical Medium Minor / Informative

Severity	Code	Description	Status
●	RFVD	Redundant Fee Variable Declarations	Unresolved
●	MCM	Misleading Comment Messages	Unresolved
●	IDI	Immutable Declaration Improvement	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved

Table of Contents

Review

Contract Name	FruitCoins
Compiler Version	v0.8.10+commit.fc410830
Optimization	200 runs
Explorer	https://bscscan.com/address/0xba18643bd61d6030182d11c1eeef9b0a51c853d4
Address	0xba18643bd61d6030182d11c1eeef9b0a51c853d4
Network	BSC
Symbol	FCS
Decimals	18
Total Supply	100,000,000

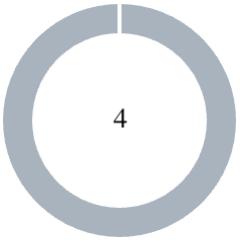
Audit Updates

Initial Audit	13 Oct 2023 https://github.com/cyberscope-io/audits/blob/main/fcs/v1/audit.pdf
Corrected Phase 2	14 Oct 2023

Source Files

Filename	SHA256
contracts/fruitcoin.sol	e2496ab2cc967eb30d852d1bebc56d51aefd584b01c3560f5224a655a54fa7f9

Findings Breakdown



Critical	0
Medium	0
Minor / Informative	4

Severity	Unresolved	Acknowledged	Resolved	Other
Critical	0	0	0	0
Medium	0	0	0	0
Minor / Informative	4	0	0	0

RFVD - Redundant Fee Variable Declarations

Criticality	Minor / Informative
Location	contracts/fruitcoin.sol#L134
Status	Unresolved

Description

The contract utilizes multiple variables, specifically `buyTaxAPercentage`, `buyTaxBPercentage`, and `buyTaxCPercentage` to apply tax on buy transactions, and `sellTaxAPercentage` and `sellTaxBPercentage` to apply fees on sell transactions. However, these variables do not contribute to the calculation of distinct values. Instead, they are simply added together to calculate the `totalTax` value. As a result, the three distinct variables for buy transactions could be merged into a single variable, and the two variables for sell transactions could also be merged into one.

```
if (isSellTransaction) {
    liquidityTaxAmount = (_value * liquidityTaxPercentage) / 100; // 3
    burnAmount = (_value * burnTax) / 100; // 1% burn  // 1
    totalTax =
        sellTaxAPercentage +
        sellTaxBPercentage +
        sellTaxCPercentage;
    taxAmount = (_value * totalTax) / 100;
}
if (isBuyTransaction) {
    liquidityTaxAmount = (_value * liquidityTaxPercentage) / 100; // 3
    totalTax =
        buyTaxAPercentage +
        buyTaxBPercentage +
        buyTaxCPercentage;
    taxAmount = (_value * totalTax) / 100;
}
```

`

Recommendation

It is recommended to combine the three different variables `buyTaxAPercentage`, `buyTaxBPercentage`, and `buyTaxCPercentage` into one variable for the buy tax.

Similarly, the two variables `sellTaxAPercentage` and `sellTaxBPercentage` could be merged into one variable for the sell tax. This consolidation will simplify the code, reduce potential points of failure, and make the contract more readable and maintainable.

MCM - Misleading Comment Messages

Criticality	Minor / Informative
Location	contracts/fruitcoin.sol#L73,94
Status	Unresolved

Description

The contract is using misleading comment messages. These comment messages do not accurately reflect the actual implementation, making it difficult to understand the source code. As a result, the users will not comprehend the source code's actual implementation.

Specifically, the contract includes a comment indicating a decrease of the fee from 10% to 7%. However, the initial value of `sellTaxAPercentage` was set to `9%` and not `10%` as indicated by the comment.

```
sellTaxAPercentage = 9;
...
function decreaseTaxPercentage() external {
        ...
        // Decrease tax percentage from 10% to 4%
        ...
        sellTaxAPercentage = 4;
    }
```

Recommendation

The team is advised to carefully review the comment in order to reflect the actual implementation. To improve code readability, the team should use more specific and descriptive comment messages. It is recommended to modify the comment to accurately reflect the initial fee. The comment should be changed from `Decrease tax percentage from 10% to 4%` to `Decrease tax percentage from 9% to 4%` to ensure clarity and accuracy in the code documentation.

IDI - Immutable Declaration Improvement

Criticality	Minor / Informative
Location	contracts/fruitcoin.sol#68,69,76
Status	Unresolved

Description

The contract declares state variables that their value is initialized once in the constructor and are not modified afterwards. The `immutable` is a special declaration for this kind of state variables that saves gas when it is defined.

```
decimals
totalSupply
deploymentTime
```

Recommendation

By declaring a variable as immutable, the Solidity compiler is able to make certain optimizations. This can reduce the amount of storage and computation required by the contract, and make it more gas-efficient.

L04 - Conformance to Solidity Naming Conventions

Criticality	Minor / Informative
Location	contracts/fruitcoin.sol#L189,198,199,200
Status	Unresolved

Description

The Solidity style guide is a set of guidelines for writing clean and consistent Solidity code. Adhering to a style guide can help improve the readability and maintainability of the Solidity code, making it easier for others to understand and work with.

The followings are a few key points from the Solidity style guide:

1. Use camelCase for function and variable names, with the first letter in lowercase (e.g., myVariable, updateCounter).
2. Use PascalCase for contract, struct, and enum names, with the first letter in uppercase (e.g., MyContract, UserStruct, ErrorEnum).
3. Use uppercase for constant variables and enums (e.g., MAX_VALUE, ERROR_CODE).
4. Use indentation to improve readability and structure.
5. Use spaces between operators and after commas.
6. Use comments to explain the purpose and behavior of the code.
7. Keep lines short (around 120 characters) to improve readability.

```
address _to
uint256 _value
address _from
```

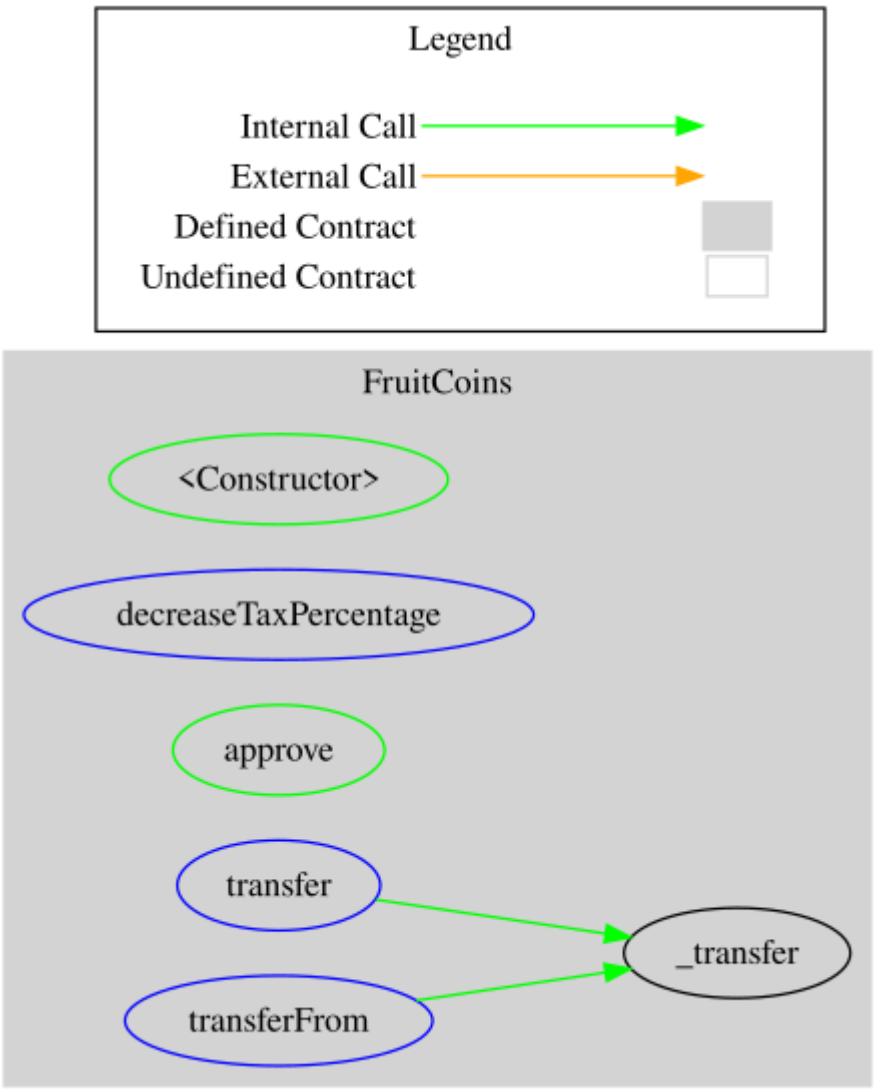
Recommendation

By following the Solidity naming convention guidelines, the codebase increased the readability, maintainability, and makes it easier to work with.
Find more information on the Solidity documentation
https://docs.soliditylang.org/en/v0.8.17/style-guide.html#naming-convention.

Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
FruitCoins	Implementation			
		Public	✓	-
	decreaseTaxPercentage	External	✓	-
	_transfer	Internal	✓	
	approve	Public	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-

Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
FruitCoins
<Constructor>
decreaseTaxPercentage
approve
transfer
_transfer
transferFrom

Summary

Fruitcoins contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements. Fruitcoins is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions. There is a limit of max 15% buy fees and 25% sell fees. The contract implements a mechanism to reduce the sell fees to 16% once 30 days have passed after the contract's deployment. However, it should be noted that this mechanism can only be activated manually (the `addressA` has to call the `decreaseTaxPercentage` function).

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. 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 Cyberscope's prior written consent. 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 regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document 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 report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment 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.

Blockchain technology and cryptographic assets present a high level of ongoing risk Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

https://www.cyberscope.io