ASSURE DEFI®
THE VERIFICATION GOLD STANDARD

Security Assessment:
CryptoCoin TOKEN

January 23, 2025

- Audit Status: **Pass**
- Audit Edition: **Advance**

VERIFIED BY ASSURE DEFI®
INTEGRITY • TRUST • CREDIBILITY

Risk Analysis

Classifications of Manual Risk Results

Classification	Description
🔴 Critical	Danger or Potential Problems.
🟠 High	Be Careful or Fail test.
🟡 Medium	Pass, Not-Detected or Safe Item.
🟢 Low	Function Detected

Manual Code Review Risk Results

Contract Privilege	Description
🟢 Buy Tax	0%
🟢 Sale Tax	0%
🟢 Cannot Buy	Pass
🟢 Cannot Sale	Pass
🟢 Max Tax	0%
ℹ Modify Tax	Yes
🟢 Fee Check	Pass
🟢 Is Honeypot?	Not Detected
🟢 Trading Cooldown	Not Detected
🟢 Can Pause Trade?	Pass
🟢 Pause Transfer?	Not-Detected
🟢 Max Tx?	Pass
🟢 Is Anti Whale?	Not-Detected
🟢 Is Anti Bot?	Not-Detected

Contract Privilege	Description
🟢 Is Blacklist?	Not-Detected
🟢 Blacklist Check	Pass
🟢 is Whitelist?	Detected
🟡 Can Mint?	Fail
🟢 Is Proxy?	Not Detected
🟢 Can Take Ownership?	Not Detected
🟢 Hidden Owner?	Not-Detected
ℹ Owner	0x515516eb8437BF4A3A49af49ccADec540bfD7875
🟢 Self Destruct?	Not Detected
🟢 External Call?	Not-Detected
🟢 Other?	Not Detected
🟢 Holders	3
🟡 Auditor Confidence	Medium
🟡 KYC Present	No
🟡 KYC URL	

The following quick summary it's added to the project overview; however, there are more details about the audit and its results. Please read every detail.

Project Overview

Token Summary

Parameter	Result
Address	0xed7257B255ED506864Cb220744CE3a381Ff37a8e
Name	CryptoCoin
Token Tracker	CryptoCoin (CrCoin)
Decimals	18
Supply	99,900,000,000,000
Platform	ETHEREUM
compiler	v0.8.17+commit.8df45f5f
Contract Name	DefiV5Token
Optimization	Yes with 200 runs
LicenseType	MIT
Language	Solidity
Codebase	https://etherscan.io/address/0xed7257B255ED506864Cb220744CE3a381Ff37a8e#code
Payment Tx	Corporate

Main Contract Assessed
Contract Name

Name	Contract	Live
CryptoCoin	0xed7257B255ED506864Cb220744CE3a381Ff37a8e	Yes

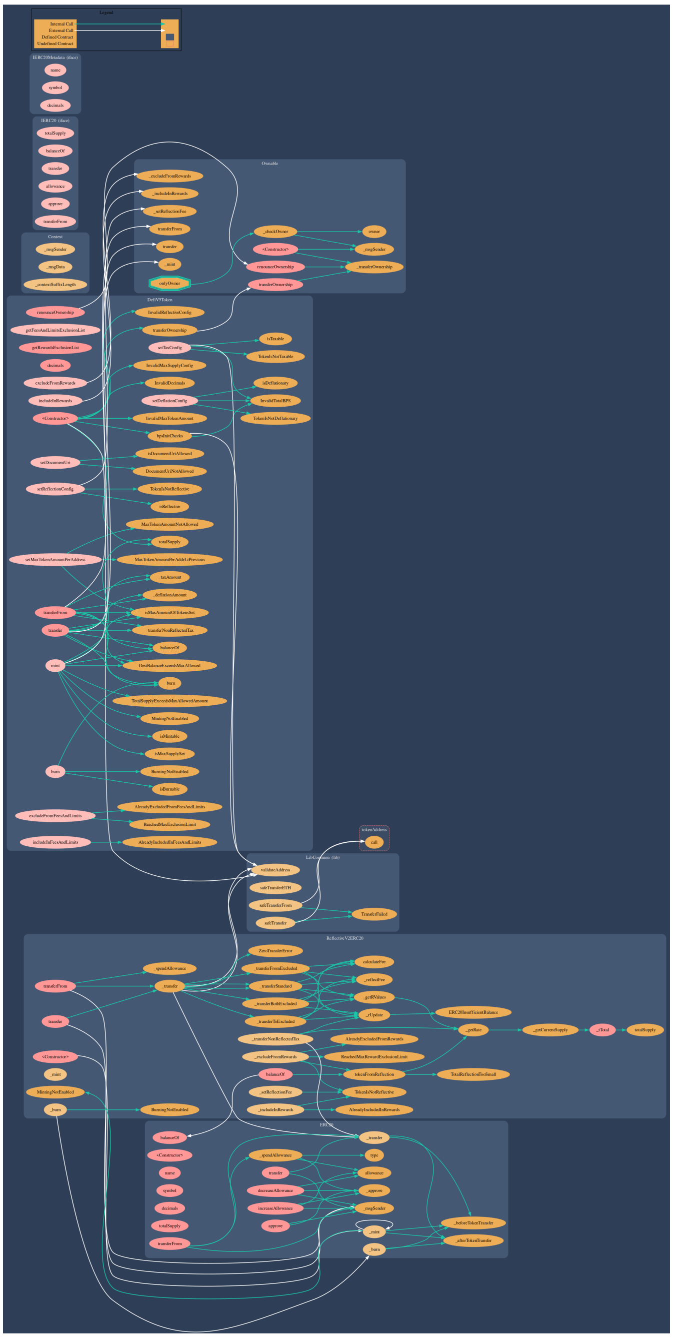
TestNet Contract was Not Assessed

Solidity Code Provided

SolID	File Sha-1	FileName
DefiV5Token	7c88fef5676abc334ad40715ccaad3de4ef70e93	DefiV5Token.sol
DefiV5Token		.sol
DefiV5Token		.sol
DefiV5Token		.sol
DefiV5Token		.sol
DefiV5Token		.sol

Call Graph

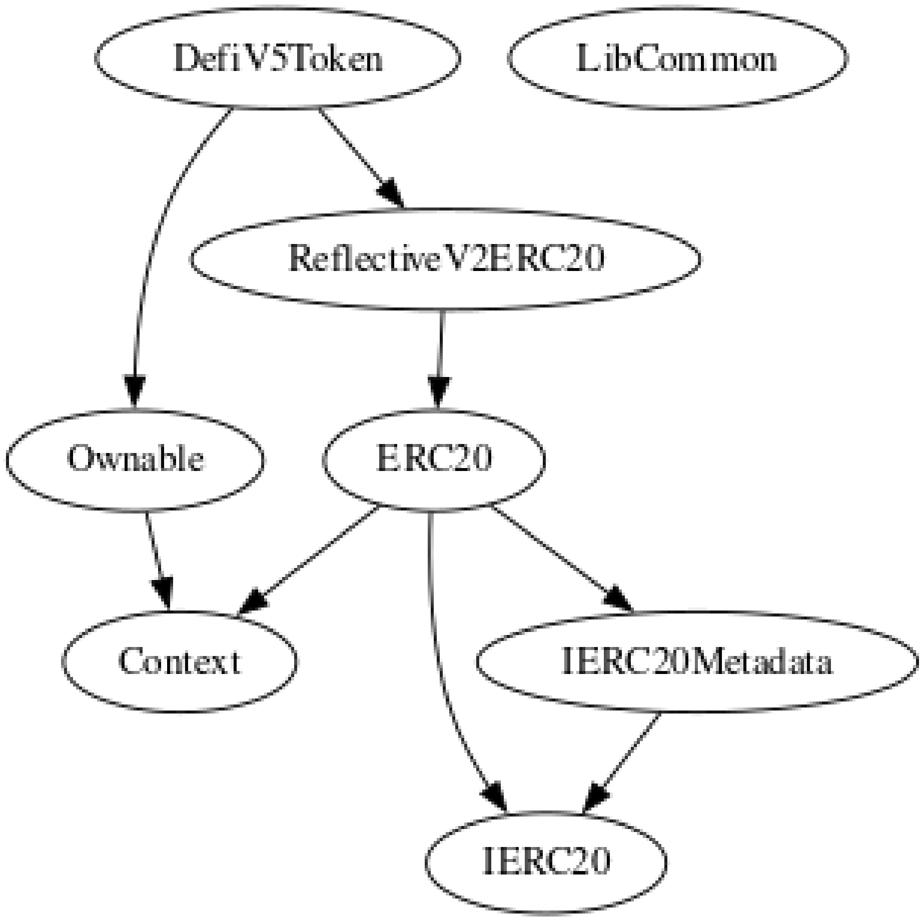
The contract for CryptoCoin has the following call graph structure.



Inheritance

The contract for CryptoCoin has the following inheritance structure.

The Project has a Total Supply of 99,900,000,000,000



CrCoin-03 | Lack of Input Validation.

Category	Severity	Location	Status
Volatile Code	🟢 Low	DefiV5Token.sol: L: 259 C: 12, L: 270 C: 12, L: 287 C: 12, L: 303 C: 12, L: 321 C: 12, L: 396 C: 12, L: 417 C: 12, L: 426 C: 12, L: 433 C: 12, L: 440 C: 12, L: 459 C: 12, L: 466 C: 12, L: 486 C: 12	Detected

Description

The given input is missing the check for the non-zero address.

The given input is missing the check for the onlyOwners need to be revisited for require..

Remediation

We advise the client to add the check for the passed-in values to prevent unexpected errors as below:

```
    ...
     require(receiver != address(0), "Receiver is the zero address");
    ...
    ...
    require(value X limitation, "Your not able to do this function");
   ...
```

We also recommend customer to review the following function that is missing a required validation. onlyOwners need to be revisited for require..

CrCoin-05 | Missing Event Emission.

Category	Severity	Location	Status
Volatile Code	🟢 Low	DefiV5Token.sol: L: 396 C: 12, L: 417 C: 12, L: 433 C: 12, L: 486 C: 12	Detected

Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.The linked code does not create an event for the transfer.

Remediation

Emit an event for critical parameter changes. It is recommended emitting events for the sensitive functions that are controlled by centralization roles.

CrCoin-19 | Centralization Privileges of.

Category	Severity	Location	Status
	● Medium	DefiV5Token.sol: L: 0 C: 14	Detected

Description

Centralized Privileges are found on the following functions.

Remediation

Inheriting from Ownable and calling its constructor on yours ensures that the address deploying your contract is registered as the owner. The onlyOwner modifier makes a function revert if not called by the address registered as the owner.

Project Action

Technical Findings Summary

Classification of Risk

Severity	Description
🔴 Critical	Risks are those that impact the safe functioning of a platform and must be addressed before launch. Users should not invest in any project with outstanding critical risks.
🟠 High	Risks can include centralization issues and logical errors. Under specific circumstances, these major risks can lead to loss of funds and/or control of the project.
🟡 Medium	Risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform
🟢 Low	Risks can be any of the above but on a smaller scale. They generally do not compromise the overall integrity of the Project, but they may be less efficient than other solutions.
ℹ Informational	Errors are often recommended to improve the code's style or certain operations to fall within industry best practices. They usually do not affect the overall functioning of the code.

Findings

Severity	Found	Pending	Resolved
🔴 Critical	0	0	0
🟠 High	0	0	0
🟡 Medium	1	1	0
🟢 Low	2	2	0
ℹ Informational	0	0	0
Total	3	3	0

Social Media Checks

Social Media	URL	Result
Twitter	https://x.com/RealCryptoCoin	Pass
Other	https://keybase.io/goofwear	Pass
Website	https://officialcryptocoin.net/	Pass
Telegram	https://t.me/goofwear1	Pass

We recommend to have 3 or more social media sources including a completed working websites.

Social Media Information Notes:

Auditor Notes: undefined

Project Owner Notes:

ASSURE™ DEFI

OFFICIAL PARTNER

Assessment Results

Score Results

Review	Score
Overall Score	85/100
Auditor Score	85/100
Review by Section	Score
Manual Scan Score	23
Auto Scan Score	37
Advance Check Score	25

The Following Score System Has been Added to this page to help understand the value of the audit, the maximum score is 100, however to attain that value the project most pass and provide all the data needed for the assessment. Our Passing Score has been changed to 84 Points for a higher standard, if a project does not attain 85% is an automatic failure. Read our notes and final assessment below.

Audit Passed

VERIFIED BY ASSURE DEFI

INTEGRITY ★ TRUST ★ CREDIBILITY

Assessment Results

Important Notes:

- Access Control: Ensure onlyOwner functions are correctly restricted to prevent unauthorized access. Verify that ownership transfer and renouncement mechanisms are secure and cannot be exploited.ı

- Immutable Variables: Confirm that immutable variables like initialSupply, initialTokenOwner, etc., are correctly initialized and used as intended.ı

- BPS Calculations: Validate that the BPS calculations for tax, deflation, and reflection do not exceed MAX_ALLOWED_BPS to prevent excessive fees. Ensure that _taxAmount and _deflationAmount calculations are accurate.ı

- Exclusion List Management: Check that the feesAndLimitsExcluded and rewardsExcluded lists are properly managed and cannot be manipulated. Confirm that MAX_EXCLUSION_LIMIT is enforced to prevent excessive exclusions.ı

- Reflection Logic: Review the reflection logic to ensure that rewards are distributed correctly and fairly among token holders.ı

- Arithmetic Operations: Although Solidity 0.8+ handles overflow/underflow, double-check arithmetic operations for correctness.ı

- External Calls: Validate the use of external libraries (LibCommon, ReflectiveV2ERC20) to ensure they are secure and correctly integrated.

- Custom Errors: Ensure custom errors are used effectively for gas optimization and clear error reporting.

- State Variables: Verify that state variables are initialized and updated correctly throughout the contract lifecycle. Pay special attention to variables like taxAddress, taxBPS, deflationBPS, and maxTokenAmountPerAddress.

- Event Emission: Confirm that events are emitted appropriately to track changes in contract state, such as DocumentUriSet, TaxConfigSet, and DeflationConfigSet.

- Minting and Burning: Ensure that minting and burning functionalities are correctly restricted to the owner and that they adhere to the constraints set by isMintable and isBurnable.

- Gas Optimization: Review the contract for potential gas optimizations, such as minimizing storage reads/writes and using efficient data structures.

- Testing and Coverage: Ensure comprehensive testing, including edge cases for all functionalities, especially around tax, deflation, and reflection mechanisms.

- Documentation: Verify that the contract is well-documented, with clear explanations of [illegible]on and its intended use.

VERIFIED BY ASSURE DEFI ™
INTEGRITY ★ TRUST ★ CREDIBILITY

Aud[illegible]85
Au[illegible]d

Appendix

Finding Categories

Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that actagainst the nature of decentralization, such as explicit ownership or specialized access roles incombination with a mechanism to relocate funds.

Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimalEVM opcodes resulting in a reduction on the total gas cost of a transaction.

Logical Issue

Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on howblock.timestamp works.

Control Flow

Control Flow findings concern the access control imposed on functions, such as owner-only functionsbeing invoke-able by anyone under certain circumstances.

Volatile Code

Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that mayresult in a vulnerability.

Coding Style

Coding Style findings usually do not affect the generated byte-code but rather comment on how to makethe codebase more legible and, as a result, easily maintainable.

Inconsistency

Inconsistency findings refer to functions that should seemingly behave similarly yet contain different code,such as a constructor assignment imposing different require statements on the input variables than a setterfunction.

Coding Best Practices

ERC 20 Conding Standards are a set of rules that each developer should follow to ensure the code meet a set of creterias and is readable by all the developers.

Disclaimer

Assure Defi has conducted an independent security assessment to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the reviewed code for the scope of this assessment. This report does not constitute agreement, acceptance, or advocation for the Project, and users relying on this report should not consider this as having any merit for financial advice in any shape, form, or nature. The contracts audited do not account for any economic developments that the Project in question may pursue, and the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude, and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are entirely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment advice, nor should it be used to signal that any persons reading this report should invest their funds without sufficient individual due diligence, regardless of the findings presented. Information is provided 'as is, and Assure Defi is under no covenant to audited completeness, accuracy, or solidity of the contracts. In no event will Assure Defi or its partners, employees, agents, or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions or actions with regards to the information provided in this audit report.

The assessment services provided by Assure Defi are subject to dependencies and are under continuing development. You agree that your access 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 with high levels of technical risk and uncertainty. The assessment reports could include false positives, negatives, and unpredictable results. The services may access, and depend upon, multiple layers of third parties.

ASSURE DEFI™
THE VERIFICATION **GOLD STANDARD**