SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS OF:
UFX TOKEN

11th March 2022

TABLE OF CONTENTS

DISCLAIMER

The information provided on this analysis document is only for general information and should not be used as a reason to invest. Coinlens Team will take no payment for manipulating the results of this audit. The score and the result will stay on this project page information on our website https://coinlens.net. Coinlens Team does not guarantee that a project will not sell off team supply, or any other scam strategy (RUG or Honeypot etc.)

NOTE

Coinlens received an application for smart contract security audit of UFX TOKEN on March 11, 2022, from the project team to discover if any vulnerability in the source code of the UFX TOKEN project as well as any contract dependencies. Standard tests have been performed using Static Analysis and Manual Review techniques.

The auditing process focuses to the following considerations with collaboration of an expert team:

- Functionality test of the Smart Contract to determine if proper logic has been followed throughout the whole process.

- Manually detailed examination of the code line by line by experts.

- Live test by multiple clients using Testnet.

- Analyzing failure preparations to check how the Smart Contract performs in case of any bugs and vulnerabilities.

- Checking whether all the libraries used in the code are on the latest version.

- Analyzing the security of the on-chain data.

PROJECT SUMMARY

Project Name: UFX TOKEN

Website: https://www.unitedfarmers.finance

Twitter: https://twitter.com/UnitedFarmersX

Telegram: https://t.me/united_farmers_x

Platform: Binance Smart Chain Token Type BEP20

Language: Solidity

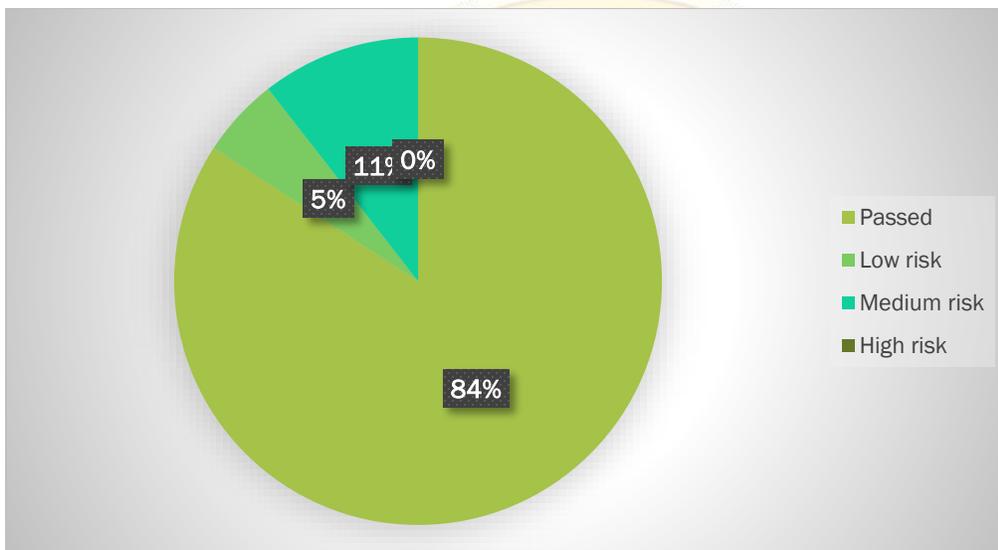Contract address: 0x119e4c3808Cf6C00E9471042F0d8B7b3d090EEe8

Audit Result:

✓ UFX TOKEN has successfully PASSED the smart contract audit with MEDIUM level severity issues.

SECURITY SCORE
84

Audit Result:	**Passed**
Ownership:	**Not renounced yet**
KYC Verification:	**N/A at the date of this review**
Audit Date:	**March 11, 2022**
Audit Team:	**COINLENS**

OVERVIEW

This Audit Report mainly focuses on overall security of UFX TOKEN Smart Contract.

CoinLens team scanned the contract and assessed overall system architecture and the smart contract codebase against vulnerabilities, exploitations, hacks, and back-doors to ensure its reliability and correctness.

Auditing Approach and Applied Methodologies

Coinlens team has performed rigorous test procedures of the project:

- Code design patterns analysis in which smart contract architecture is reviewed to ensure it is structured according to industry standards and safe use of third-party smart contracts and libraries.

- Line-by-line inspection of the Smart Contract to find any potential vulnerability like race conditions, transaction-ordering dependence, timestamp dependence, and denial of service attacks.

- Unit testing Phase, we conducted custom unit tests written for each function in the contract to verify that each function works as expected.

- Automated Test performed with our in-house developed tools to identify vulnerabilities and security flaws of the Smart Contract.

The focus of the audit was to verify that the Smart Contract System is secure, resilient, and working according to the specifications. The audit activities can be grouped in the following three categories:

Security

Identifying security related issues within each contract and the system of contract.

Code Correctness and Quality.

A full review of the contract source code. The primary focus included:

- ☻ Accuracy
- ☻ Readability
- ☻ Sections of code with high complexity
- ☻ Quantity and quality of test coverage

Risk Classification Vulnerabilities are classified in 3 main levels as below based on possible effect to the contract.

High level vulnerability Vulnerabilities on this level must be fixed immediately as they might lead to fund and data loss and open to manipulation. Any High-level finding will be highlighted with **RED** text.

Medium level vulnerability Vulnerabilities on this level also important to fix as they have potential risk of future exploit and manipulation. Any Medium-level finding will be highlighted with **ORANGE** text.

Low level vulnerability Vulnerabilities on this level are minor and may not affect the smart contract execution. Any Low-level finding will be highlighted with **BLUE** text

Vulnerability Checklist

No.	Description	Result
a)	Compiler warnings.	**Passed**
b)	Race conditions and reentrancy. Cross-function race Conditions.	**Passed**
c)	Possible delays in data delivery.	**Passed**
d)	Fallback function security.	**Passed**
e)	Safe Zeppelin module.	**Passed**
f)	Cross-function race conditions.	**Passed**
g)	Oracle calls.	**Passed**
h)	DoS with Revert.	**Passed**
i)	Timestamp dependence.	**Passed**
j)	Integer Overflow and Underflow.	**Passed**
k)	Private user data leaks.	**Passed**
l)	Malicious Event log.	**Passed**
m)	Economy model.	**Passed**
n)	Arithmetic accuracy.	**Passed**

UFX TOKEN Smart Contract Audit

Contract Analysis

Description	Status	Notes
No prior similar token contracts	✓	
Source does not contain a mint function	✗ Low risk	
Ownership renounced or source does not contain an owner contract	✗ Low risk	[Ownership to be renounced to Masterchef once contract is launched].
Verified contract source	✓	
Token is sellable (not a honeypot) at this time	✓	

Description	Status
Automatic LP is going to Owner wallet	✗
Owner can mint more coins	✓ Low risk
Contract owner can renounce ownership	✓
Contract owner can transfer ownership	✓
Contract owner can exclude an address from transactions.	✓
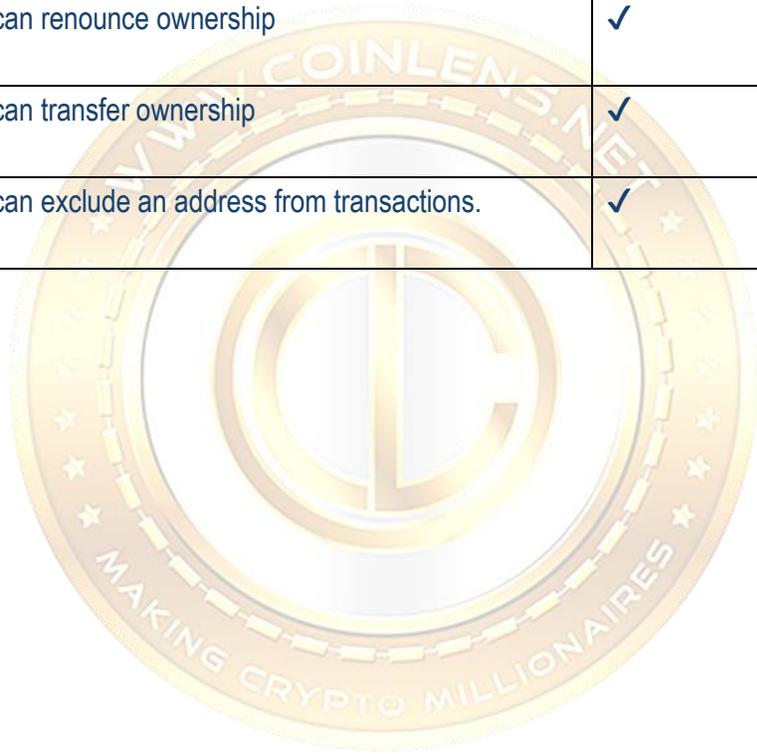
For this section the code was tested/read line by line by our developers. Additionally, Remix IDE's JavaScript VM and Kovan networks used to test the contract functionality.

Automated Audit

Remix Compiler Warnings

It throws warnings by Solidity's compiler. No issues found.

Recommendations.

Much as there is an anti-whale feature, I recommend another Protection feature. Antisniper. Like for exampel, buying too fast will mark you as a sniper, and you will be unable to sell your tokens or transfer them in any way. This includes tools and other non-hand methods. There is no amnesty. There should also be a new special algorithm for snipers, that shouldn't catch normal buyers. Only one trade (on a per user basis) can be made per block. If a single person tries to make two trades in the same block, the second will revert. This should include a maximum gas limit, setting gwei too high may as well reverts the transaction.

Important Notice for Investors

As Coinlens team we are mainly auditing the contract code to find out how it will be functioning, and risks which are hidden in the code if any. There are many factors that must be taken into consideration before investing to a project, like:

- ☺ Ownership status

- ☺ Project team approach

- ☺ Marketing

- ☺ General market condition

- ☺ Liquidity

- ☺ Token holdings etc.

Investors must always do their own research and manage their risk considering different factors which can affect the success of a project.