DcentraLab Diligence

dcentralab.com/diligence

TokensFarm

UniV3 Audit

TokensFarm

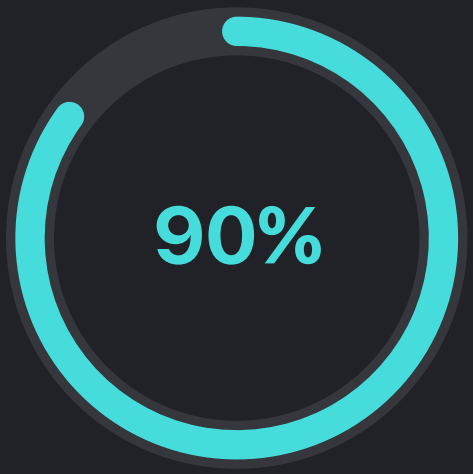
https://www.tokensfarm.com

Provided By DcentraLab Diligence on June 19, 2023

Security Audit Score

Pass

DcentraLab Diligence team has conducted an extensive audit on TokensFarm's UniV3 Contracts and has found the code to be in low risk level given proper deployment and multi-sig permissioning.



- Low Risk
- Small Risk
- Medium Risk
- High Risk

Scope

Audited Repository:

https://github.com/Tokensfarm/tokensfarm-contracts-v2

Audited Branch:

develop

Audited Commit Hash:

0165e18640ce0bc6cb192353577d9991c5d66fb1

Audited Contracts:

TokensFarmUniV3.sol
UniswapUtils.sol
UniswapV3Staker.sol
TokensFarmUpgradable.sol

Risks:

DcentraLab Diligence (DD) has performed all checks and verifications in its capacity to ascertain the safety of the code. However, it should be noted that misuse of the code, bad deployment practices, bad key management, exposing of private keys of the deployer and/or owner address and/or multi-sig signer addresses and/or fee collector address and/or any exposition of the code to malicious actors may result in an exploit of the code and loss of state and/or funds. Furthermore, there is always a chance that other Smart Contracts code could be written and deployed to cause the provided code by DD to act outside the intended scope by the client, to the point of causing state corruption or loss of funds to the client of the users of the code.

Issues Severity Reference Table

Type

Informational

This issue is not critical and does not pose an immediate threat to the functionality or security of the smart contract. It is simply an informational item that the auditors have identified and recommends addressing for best practices or to improve the overall performance of the contract.

Low

This issue is relatively minor and does not pose a significant risk to the functionality or security of the smart contract. While it is recommended to address these issues to ensure the highest level of quality and security, they are not likely to cause significant problems if left unaddressed.

Medium

This issue poses a moderate risk to the functionality or security of the smart contract. While it may not be immediately exploitable, it has the potential to cause problems in the future if left unaddressed. It is recommended to address these issues as soon as possible to prevent any potential negative impact on the contract.

High

This issue poses a significant risk to the functionality or security of the smart contract. Addressing these issues as soon as possible is recommended to prevent any potential negative impact on the contract. Failure to address these issues could result in significant problems and potential loss of funds or other assets.

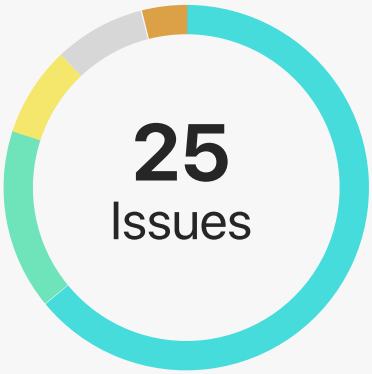
Critical

This issue poses an immediate and severe risk to the functionality or security of the smart contract. It is recommended to address these issues immediately to prevent any potential negative impact on the contract. Failure to address these issues could result in catastrophic problems and significant loss of funds or other assets.

Discussion

The issue severity is dependent on design, centralization, and product specifications of the project.

Findings Summary



- Informational
- Low Risk
- Medium Risk
- High Risk
- Critical Risk
- Discussion

ID	Title	Severity	Status
1	TokensFarmUpgradeable is not upgradeable	Medium	Acknowledged
2	Missing events	Low	Acknowledged
3	Large revert messages	Informational	Acknowledged
4	Repeating require statements	Informational	Acknowledged
5	Contract should be marked as abstract	Informational	Acknowledged
6	Improper function visibility	Low	Acknowledged
7	Inheritance of non-upgradeable contracts	Discussion	Acknowledged
8	Missing error messages	Medium	Acknowledged
9	Redundant SafeMath usage	Informational	Acknowledged
10	Redundant SafeMath usage	Informational	Acknowledged
11	Messy codebase	Discussion	Acknowledged

Findings Summary

ID	Title	Severity	Status
12	Redundant storage load	Informational	Resolved
13	Redundant on-chain computation	Informational	Resolved
14	Optimizable castings	Informational	Resolved
15	Optimizable castings	Informational	Resolved
16	edundant uint check	Low	Resolved
17	Redundant address check	Low	Acknowledged
18	Double computation of 'incentiveId'	Informational	Resolved
19	Forwarding 'key' as an argument where it's not needed	Informational	Partially Resolved
20	Invalid reward computation	High	Resolved
21	Optimizable mapping value setting	Informational	Resolved
22	Redundant global variable	Informational	Resolved
23	Repeating 'require' statement	Informational	Resolved
24	Repeating statement	Informational	Acknowledged
25	Optimizable 'require statement	Informational	Acknowledged

Complete Analysis

General Notices:

- Resolving Issue ID 19 will reduce gas consumption and contract size in a non-negligible manner.
- Missing events and revert messages can lead to hardships during the debugging process.
- Having no properly working tests is significantly reducing the auditor's issue finding capabilities.
- Make sure to perform thorough testing of audited code before deployment to production.

ID 1: **Status: Acknowledged**

Medium | TokensFarmUpgradeable is not upgradeable

Present at: TokensFarmUpgradeable.

Description: Even though the logic of this contract could be changed, it is missing the gap which would enable developers to add new variables over time. If new variables are added to the current code that would cause an issue in the entire storage of any contract which inherits TokensFarmUpgradeable.

Fix feedback: Acknowledged

ID 2: **Status: Acknowledged**

Low | Missing events

Present at: TokensFarmUpgradeable

Description: Important setters emit no events.

Complete Analysis

Recommendation: Consider adding events to the important setters.

Fix feedback: Acknowledged

ID 3: **Status: Acknowledged**

Informational | Large revert messages

Present at: TokensFarmUpgradeable

Description: Revert messages are unnecessarily large.

Recommendation: Consider reducing the size of revert messages.

Fix feedback: Acknowledged

ID 4: **Status: Acknowledged**

Informational | Repeating require statements

Present at: TokensFarmUpgradeable

Description: Same 'require' statements are being repeated through the code.

Recommendation: Consider making internal functions with such 'require' statements and call them in needed places instead.

Fix feedback: Acknowledged

Complete Analysis

ID 5: **Status: Acknowledged**

Informational | Contract should be marked as abstract

Present at: TokensFarmUpgradeable

Description: Contract is not meant to be deployed by itself so it can be considered abstract.

Recommendation: Consider marking the contract as abstract.

Fix feedback: Acknowledged

ID 6: **Status: Acknowledged**

Low | Improper function visibility

Present at: TokensFarmUpgradeable

Description: Contract contains internal function 'setCongressAndMaintainersRegistry()' which can be accessed at any time through the logic of the contract that inherits TokensFarmUpgradeable. This is not an issue of great risk but it could cause unintentional state changes by a bug introduced along the way.

Recommendation: We recommend that you use private functions instead of internal for the state changes and access such functions only through authorized ways or initialization flow.This way you limit state changes to the specific predefined set of rules, which make your architecture more secure and future proof to bugs.

Fix feedback: Acknowledged

Complete Analysis

ID 7: **Status: Acknowledged**

Discussion | Inheritance of non-upgradeable contracts

Present at: TokensFarmUniV3

Description: Contract inherits non-upgradeable contracts.

Recommendation: We recommend you to carefully review the storage layout which is constructed upon the existing inheritance scheme and pay attention to the contract changes that might come up in the future in order to avoid the potential storage related issues such as data slot shifting/overriding or other.

Fix feedback: Acknowledged

ID 8: **Status: Acknowledged**

Medium | Missing error messages

Present at: TokensFarmUniV3

Description: Contract has a visible lack of revert/error messages on its 'require' statements.

Recommendation: We recommend you to add error messages in order to avoid issues with debugging and provide better UX.

Fix feedback: The contract already has 21KB, 24KB is the limit so we will proceed without require messages, tenderly is a debugger that we can rely on. Acknowledged

Complete Analysis

ID 9: **Status: Acknowledged**

Informational | Redundant SafeMath usage

Present at: TokensFarmUniV3 @ L404-405

Description: SafeMath 'sub()' function is used even though values have already been checked previously (lines 395-398).

Recommendation: We recommend using pure subtraction instead of SafeMath in the mentioned place.

Fix feedback: We are using safe math everywhere regardless if we know that it won't come to an underflow or overflow. Acknowledged

ID 10: **Status: Acknowledged**

Informational | Redundant SafeMath usage

Present at: TokensFarmUniV3 @ L533-535

Description: SafeMath 'sub()' function is used even though values have already been checked previously (lines 509-512).

Recommendation: We recommend using pure subtraction instead of SafeMath in the mentioned place.

Fix feedback: We are using safe math everywhere regardless if we know that it won't come to an underflow or overflow. Acknowledged

Complete Analysis

ID 11: **Status: Acknowledged**

Discussion | Discussion: Messy codebase

Present at: Throughout the repository

Description: Code present on repository is not formatted properly.

Recommendation: Consider applying lint and prettifying the code.

Fix feedback: Acknowledged

ID 12: **Status: Resolved**

Informational | Redundant storage load

Present at: TokensFarmUniV3 @ L755

Description: Variable emitted in an event is loaded from storage while there's a local version of it parsed into a function as an argument.

Recommendation: Consider emitting a local version of the variable instead of loading it from storage.

Fix feedback: Fixed as suggested

ID 13: **Status: Resolved**

Informational | Redundant on-chain computation

Present at: TokensFarmUniV3

Complete Analysis

Description: Function 'incentiveId' is performing computation based on the 'key' structure parsed as an argument. This structure is large and parsing it to the contract should be evaded unless necessary.

Recommendation: Consider parsing '_incentiveId' (key hash) directly to the contract in functions where the key itself is not needed (such as 'setFlatFees'). This should help reduce bytecode size and tx execution expenses.

Fix feedback: Fixed as suggested

ID 14: **Status: Resolved**

Informational | Optimizable castings

Present at: TokensFarmUniV3 @ L452-467

Description: Inside the function '_checkIfIncentiveCanBeCreated', 'key.pool' value is casted to address multiple times.

Recommendation: Consider optimizing code by making only one single casting and saving that value to use it throughout the flow.

Fix feedback: Fixed as suggested

ID 15: **Status: Resolved**

Informational | Optimizable castings

Present at: TokensFarmUniV3 @ L594-618

Description: Inside the function 'endIncentive', 'key.pool' value is casted to address multiple times.

Complete Analysis

Recommendation: Consider optimizing code by making only one single casting and saving that value to use it throughout the flow.

Fix feedback: Fixed as suggested

ID 16: **Status: Resolved**

Low | Redundant uint check

Present at: TokensFarmUniV3 @ L665

Description: At mentioned line, there is a 'require' statement checking if uint value is greater than or equal to zero.

Recommendation: Consider removing the check.

Fix feedback: Fixed as suggested

ID 17: **Status: Acknowledged**

Low | Redundant address check

Present at: TokensFarmUniV3 @ L1291 & L394 & L877 & 1359

Description: At mentioned line, msg.sender is indirectly checked not to equal zero address.

Recommendation: Consider removing the check.

Fix feedback: Requires are there in case we want to change who is going to be the user in the function and from where that address can come. Acknowledged

Complete Analysis

ID 18: **Status: Resolved**

Informational | Double computation of 'incentiveId'

Present at: TokensFarmUniV3 @ L1241-1268

Description: Function '_unstake' contains two computations of 'incentiveId'.

Recommendation: Consider computing 'incentiveId' only once and then reuse that value throughout the flow.

Fix feedback: Fixed as suggested

ID 19: **Status: Partially Resolved**

Informational | Forwarding 'key' as an argument where it's not needed

Present at: TokensFarmUniV3

Description: Many functions accept the 'key' (sizable struct), while they only need 'incentiveId'.

Related Issues: ID 13, ID 18

Recommendation: Consider optimizing the flow in such a way that you don't pass the 'key' argument when it is not necessary.

Fix feedback: For all function where whole key was not necessary rather only hash of the key we fixed it as suggested.

Complete Analysis

ID 20: **Status: Resolved**

High | Invalid reward computation

Present at: TokensFarmUniV3 @ L1201-1232

Description: Function '_unstakeAndTransferRewardsFromV3StakerToFarm' computes pending amount by subtracting rewards amount from before token 'unstake' from rewards amount after the mentioned action. This results in the amount which is just a fraction of the user's rewards.

Recommendation: Retrieve rewards only after the 'unstake' and call claimReward with that value as an argument.

Fix feedback: Fixed as suggested

ID 21: **Status: Resolved**

Informational | Optimizable mapping value setting

Present at: TokensFarmUniV3 @ L1267

Description: At the mentioned line there is a statement setting a 'false' value to the mapping which points to a boolean. That is also a default value, so a cheaper option would be to use 'delete'.

Recommendation: Consider using 'delete' instead of setting value to 'false' manually.

Fix feedback: Fixed as suggested

Complete Analysis

ID 22: **Status: Resolved**

Informational | Redundant global variable

Present at: TokensFarmUniV3

Description: Variable 'noOfNFTsStakedInIncentive' is tracked and updated throughout the flow even though same value can be accessed at all times by calling 'stakedNFTsIntoIncentive[incentiveId].length'

Recommendation: Consider optimizing the flow by using 'stakedNFTsIntoIncentive[incentiveId].length' instead of 'noOfNFTsStakedInIncentive'.

ID 23: **Status: Resolved**

Informational | Repeating 'require' statement

Present at: TokensFarmUniV3

Description: The following statement is being repeated multiple times throughout same flow(s): 'require(tokenOwner[tokenId] == user);'

Recommendation: Consider following the flows and making sure that statement is repeated only one time in a single flow.

Fix feedback: Fixed as suggested

Complete Analysis

ID 24: **Status: Acknowledged**

Informational | Repeating statement

Present at: TokensFarmUniV3

Description: The following statement is being repeated multiple times throughout the same flow(s): 'address user = msg.sender;'.

Recommendation: Consider reviewing the flows and making sure that statement is repeated only one time in a single flow, if you already set the 'user' variable value, forward it and do not repeat the statement.

Fix feedback: Acknowledged

ID 25: **Status: Resolved**

Informational | Optimizable 'require statement'

Present at: TokensFarmUniV3 @ L1377-1384

Description: The following statement requires that 'range > 0 && value > 0 && value >= range' in place where range value has no implication if it is zero or not.

Recommendation: Consider reviewing the flow and reducing the 'require' statement to a single check: 'value => range'.

Fix feedback: Acknowledged

Disclaimer:

DcentraLab Diligence (DD) has provided the code to the client as is and assumes no responsibility nor legal liability for any use client may do with the code. Any and all usage and/or deployment of the code provided by DcentraLab Diligence will be done solely by the client, at the sole discretion, responsibility, risk, and legal liability of the Client, and DD will not be held accountable or liable for any loss of funds, security exploits or incidents, or any other unintended or negative outcome that may occur in relation to the code provided by DD.

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 DD 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 report and the provided code or services as part of the SOW pertaining to 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 it 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.

Blockchain technology and cryptographic assets present a high level of ongoing risk. DD's position is that each company and individual are responsible for their own due diligence and continuous security. DD'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 DD are subject to dependencies and are under continuing development. You agree that your access and/or use, including but not limited to any services, code, 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.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, DcentraLab Diligence (DD) HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, DD SPECIFICALLY DISCLAIMS

ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

WITHOUT LIMITING THE FOREGOING, DD MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT / VERIFICATION REPORT, WORK PRODUCT, CODE OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET THE CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE.

WITHOUT LIMITATION TO THE DISCLAIMER [ASSESSMENT NAME] FOREGOING, DD PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET THE CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR-FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER DD NOR ANY OF DD'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION, CODE OR CONTENT PROVIDED THROUGH THE SERVICE. DD WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT OR CODE, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, CODE, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS," AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN THE CUSTOMER AND THE THIRD-PARTY
OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS. THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO THE CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT DD'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS. THE REPRESENTATIONS AND WARRANTIES OF DD CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF THE CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH
REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.
FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS, CODE, OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL,
REGULATORY, OR OTHER ADVICE.

dcentralab.com/diligence

DcentraLab Diligence

Provided By DcentraLab Diligence on June 19, 2023