CERTIK

Preliminary Comments

KIKI

Nov 16th, 2021

Table of Contents

Summary

This report has been prepared for KIKI to discover issues and vulnerabilities in the source code of the KIKI project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

Overview

Project Summary

Project Name	KIKI
Platform	ethereum
Language	Solidity
Codebase	
Commit	

Audit Summary

Delivery Date	Nov 16, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

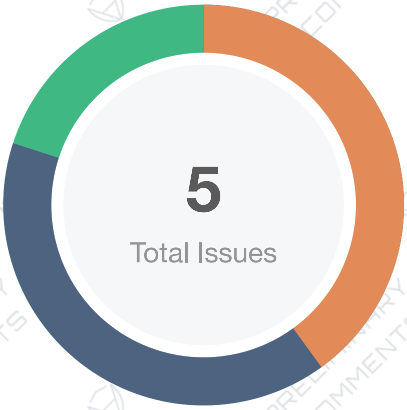
Vulnerability Summary

Vulnerability Level	Total	ⓘ Pending	⊗ Declined	ⓘ Acknowledged	ⓘ Partially Resolved	✓ Resolved
● Critical	0	0	0	0	0	0
● Major	2	2	0	0	0	0
● Medium	0	0	0	0	0	0
● Minor	0	0	0	0	0	0
● Informational	2	2	0	0	0	0
● Discussion	1	1	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
SOK	core/SafeOwnable.sol	6e80414163c87bc492caf1f92019b3e4b34b4e7f09f96199c47f8bffd2257f6f
MCK	farm/MasterChef.sol	bb2b83f39af0ece64571391cf92042471a034044f9f552d8c1b8dbea702fea1c
IER	interfaces/IERC20Mintable.sol	9df75b56f78c5af72eb9a461d1499849c7e1aefd0fe5fca78df8ed1d7cb1624f
IKI	interfaces/IKIKIVault.sol	f9ceafa656acd27d9b6f5deaf28c9fd366508a9e19b90aca7344c81d2c1616cb
IWE	interfaces/IWETH.sol	1e91de2a26d8630d9ee66015838ade8def0b3341fb64d55b3ddd760d37df6ac0
KIT	token/KIKIToken.sol	13d5d4627ebeb5082e7c03b58efb41d85ba6ca8161b22aa79387d5d2227948f5
MSK	token/MultiSignature.sol	077b486836172cd18214ca44544c8ae80e4cc57aaa6b4a92ff6a4f5b45af1bdf
TLK	token/TokenLocker.sol	870dcd5e3ade6ef5b9d0b987d4fcd5e9941636fb0509df7c2beea74a0ed96cd3
KIV	vault/KIKIVault.sol	4dfbf309e62f77a59732490b6115b2c900151e5f8d7ac98fcbd59ca175507647
TLI	vault/TeamLocker.sol	dfdf9175f89c8e9ab1111abf3800a702abbddeb58e8c7f634d5a9f96454d6b0a

Findings



🟥	**Critical**	**0** (0.00%)
🟧	**Major**	**2** (40.00%)
🟨	**Medium**	**0** (0.00%)
🟨	**Minor**	**0** (0.00%)
🟦	**Informational**	**2** (40.00%)
🟩	**Discussion**	**1** (20.00%)

ID	Title	Category	Severity	Status
KIKI-01	Centralization Risk	**Centralization / Privilege**	● **Major**	ⓘ Pending
KIV-01	Spelling Error In `updateRooHash()`	Coding Style	● Informational	ⓘ Pending
MSK-01	Centralization Risk In `MultiSignature.sol`	**Centralization / Privilege**	● **Major**	ⓘ Pending
TLI-01	Economic Model of `claim()` In `TeamLocker.sol`	Logical Issue	● Discussion	ⓘ Pending
TLK-01	Redundant Code	Logical Issue	● Informational	ⓘ Pending

KIKI-01 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	● Major	Global	ⓘ Pending

Description

In the contract `MasterChef`, the role `owner` has the authority over the following function:

- `updateMultiplier` - Set the reward multiplier.
- `updateRewardPerBlock` - Set the reward per block.
- `add` - Add a new pool.
- `set` - Set the allocation point of the pool.

In the contract `KIKIToken`, the role `owner` has the authority over the following function:

- `addMinter` - Add a new minter and set the limited amount of this minter.
- `delMinter` - Delete a minter.
- `renounceOwnership` - Renounce owner.
- `mint` - Send `KIKI` token to the minter address by himself.

In the contract `KIKIVault`, the role `owner` has the authority over the following function:

- `updateRooHash` - Set the `rootHash` and mint reward token to this contract.

Any compromise to the `owner` account may allow the hacker to take advantage of this.

Recommendation

We advise the client to carefully manage the `owner` account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., Multisignature wallets.

Indicatively, here is some feasible suggestions that would also mitigate the potential risk at the different level in term of short-term and long-term:

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;

- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key;
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

KIV-01 | Spelling Error In `updateRooHash()`

Category	Severity	Location	Status
Coding Style	● Informational	projects/KIKI/contracts/vault/KIKIVault.sol (864744b): 29	ⓘ Pending

Description

The function name has one word misspelled. The correct spelling is `updateRootHash`.

Recommendation

We recommend changing it to the correct spelling.

MSK-01 | Centralization Risk In `MultiSignature.sol`

Category	Severity	Location	Status
Centralization / Privilege	● **Major**	projects/KIKI/contracts/token/MultiSignature.sol (864744b): 83, 74, 57, 49	ⓘ Pending

Description

In the contract `MultiSignature`, the '_signaturer' role has the authority over the following function:

- [applyToken] and [acceptApplyToken] Mint the reward token to the `receiver` by vote.
- [applySetReceiver] and [acceptApplySetReceiver] Change the `receiver` address by vote.

Any compromise to the `_signaturer` account may allow the hacker to take advantage of this.

Recommendation

We advise the clients to carefully manage the `_signaturer` account's private key to avoid any potential risks of being hacked.

Indicatively, here is some feasible suggestions that would also mitigate the potential risk at the different level in term of short-term and long-term:

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

TLI-01 | Economic Model of `claim()` In `TeamLocker.sol`

Category	Severity	Location	Status
Logical Issue	● Discussion	projects/KIKI/contracts/vault/TeamLocker.sol (864744b): 54~55	ⓘ Pending

Description

The organization gets 1/9 of `vault.totalReleaseAmount` and the developer gets 2/9 of `vault.totalReleaseAmount`. Isn't that too high?

TLK-01 | Redundant Code

Category	Severity	Location	Status
Logical Issue	● Informational	projects/KIKI/contracts/token/TokenLocker.sol (864744b): 130~132	Pending

Description

In the following code:

```
125  if (receiver.totalReleaseAmount.sub(receiver.alreadyReleasedAmount) < nextReleaseAmount) {
126      nextReleaseAmount = receiver.totalReleaseAmount.sub(receiver.alreadyReleasedAmount);
127  }
128
129  alreadyReleaseAmount = receiver.alreadyReleasedAmount;
130  remainReleaseAmount = receiver.totalReleaseAmount.sub(receiver.alreadyReleasedAmount);
```

Replace the `remainReleaseAmount` variable, and this code means:

```
if (remainReleaseAmount < nextReleaseAmount) {
    nextReleaseAmount = remainReleaseAmount;
}
```

But then line 130 is redundant with the previous code:

```
130  if (nextReleaseAmount > remainReleaseAmount) {
131      nextReleaseAmount = remainReleaseAmount;
132  }
```

Recommendation

We recommend removing the redundant code.

Appendix

Finding Categories

Centralization / Privilege

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

Logical Issue

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

Coding Style

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

Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

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 CertiK'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 CertiK 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 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.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK's position is that each company and individual are responsible for their own due diligence and continuous security. CertiK'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 CertiK is subject to dependencies and 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.

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, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET 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 FOREGOING, CERTIK PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET 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 CERTIK NOR ANY OF CERTIK'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK 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 (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, 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 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 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 CERTIK'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 CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF 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 CERTIK 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 OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

About

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.