DetectBox
Powered by UNSNARL

#DECENTRALISED PROACTIVE AUDIT WAR-ROOMS

Token Contract Security Assessment

KUNJI FINANCE

JULY 5TH, 2023

TABLE OF CONTENTS

SUMMARY

This report has been prepared for Kunji Finance to discover issues and vulnerabilities in the source code of their token contract 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

CERTIFICATION

This is to Certify that the **Kunji Security Assessment Report** was prepared using gold standard web3 security standards with all standard and advanced checks deployed.

The **Main Auditor** Involved in the Audit was -
devScrooge(@devScrooge)

The Auditor mentioned hereby was chosen after self due-dilligence from the project's end.

All reported issues were either acknowledged or fixed by the project.

DetectBox's audit mitigation & residual vulnerability check did not find any major vulnerability to report within the Scope and requirements of the audit.

From
Team DetectBox

Kunji
Security Assessment Report

ABOUT DETECTBOX

DetectBox by **UNSNARL** brings to you world's first Decentralised Audit War-Rooms. An intense collaborative Environment where chosen auditors/audit team, project's decision makers, Projects developers and Detect-Wardens join hands to conduct rigorous security audits with absolute transparency followed by a **Proof of Audit (POA).**

We at DetectBox not only find bugs in smart-contracts but also take a deep look into the project's **tokenomics**, **white-paper**, **business logic**, **functional flows** and overall development health.

At DetectBox, you get to choose from a pool of Independent Security researchers which is curated by strong due-diligence. List your project requirements, verify the auditor's past work from their profiles and choose the auditor that rightly fits your project's needs and your budget.

Our auditors have successfully completed **200+ audits**, found over **2200+ vulnerabilities** and secured over **$102M+ worth of TVL.**

To know more about us visit - detectbox.io
To see our docs - https://unsnarl.gitbook.io/detectbox/

Yours Securely
UNSNARL

OVERVIEW

Project Summary

Kunji Finance aims to empower individual investors with access to active portfolio management backed by transparency, security, and community-led governance. Dedicated to creating a decentralized platform where every investor has control over their investments, and access to expert led risk managed investment strategies

Audit Summary

A time-boxed independent security assessment of the Kunji Token Contract was done by devScrooge(@devScrooge) and Team DetectBox, with a focus on the security aspects of the application's implementation.

We performed the security assessment based on the agreed scope, following our approach and methodology. Based on our scope and our performed activities, our security assessment revealed 1 Medium severity issue. Additionally, different informational and gas optimization suggestions were also made which, if resolved appropriately, may improve the quality of the Project's Smart contract.

Audit Timeline : 3rd July'23 - 5th July'23

Contract Name : MyToken.sol

OVERVIEW

Audit Methodology

During our security assessments, we uphold a rigorous approach to maintain high-quality standards. Our methodology encompasses thorough functional testing and meticulous manual code reviews. To ensure comprehensive issue coverage, we employ checklists derived from industry best practices and widely recognized concerns, specifically tailored to Solidity smart contract assessment.

Throughout the smart contract audit process, we prioritize the following aspects to uphold excellence:

1. Code Quality: We diligently evaluate the overall quality of the code, aiming to identify any potential vulnerabilities or weaknesses.
2. Best Practices: Our assessments emphasize adherence to established best practices, ensuring that the smart contract follows industry-accepted guidelines and standards.
3. Documentation and Comments: We meticulously review code documentation and comments to ensure they accurately reflect the underlying logic and expected behaviour of the contract.

Auditing smart contracts involves a comprehensive analysis of the code to identify potential vulnerabilities and security risks. To achieve comprehensive coverage, we employ a series of security checklist tables, each addressing specific areas of concern. These include:

- System / Platform
- Access Control
- Storage

OVERVIEW

- Gas Issues and Efficiency
- Code Issues
- Error Handling and Exception Handling:
- Transaction Handling
- Entrypoint Validation
- Administration and Operator Functions
- Additional Topics and Test Cases

Vulnerability Summary

Severity classification

Severity	Impact : High	Impact : Medium	Impact : Low
Likelihood : High	Critical	High	Medium
Likelihood : Medium	High	Medium	Low
Likelihood : Low	Medium	Low	Low

Findings Summary

High	0 issues
Medium	1 issue
Low	0 issues
Informational	2 issues
Gas Optimisations	0 issues

OVERVIEW

Audit Scope

The code under review is composed of a single smart contract written in Solidity language and includes 60 SLOC- solidity lines of code (only source-code lines).

Sl. No.	Lines	SLOC
contracts/vmm.py	66	60

Total SLOC : 60

Auditor Involved :

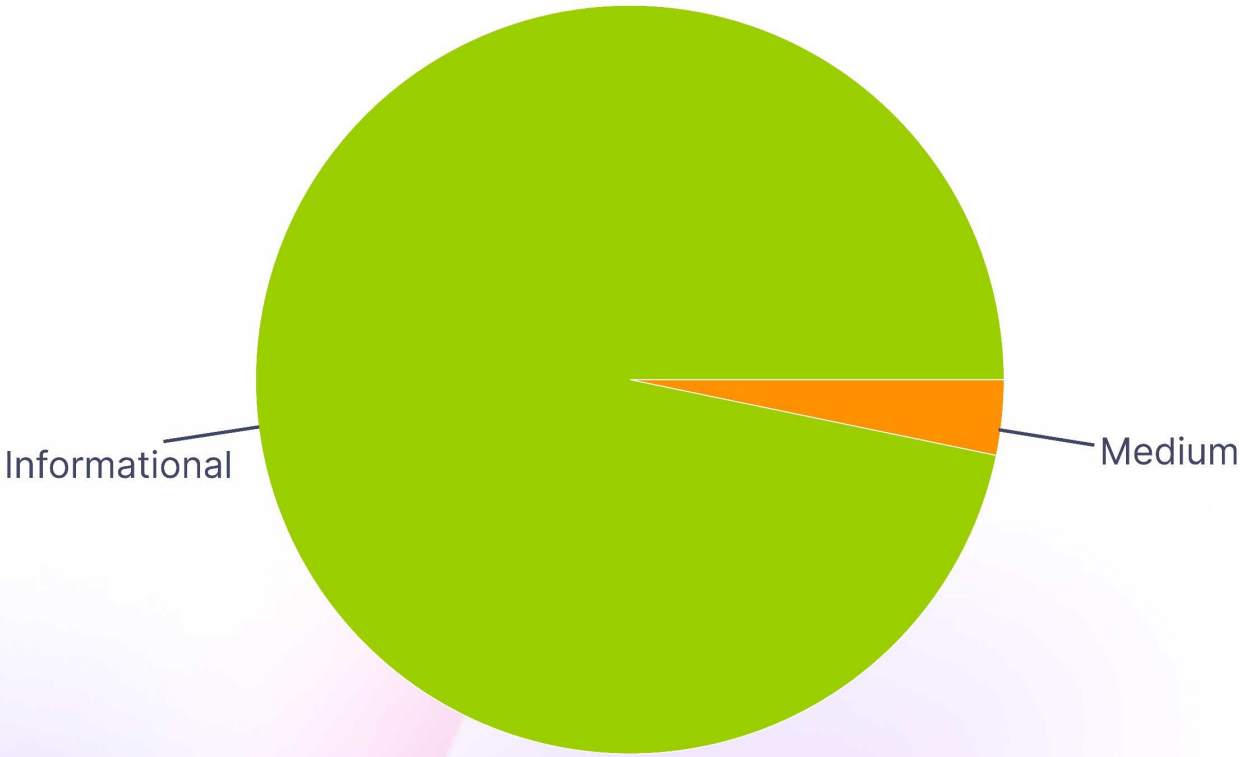
Main Auditor -

devScrooge - https://app.detectbox.io/profile/devScrooge

FINDINGS

Detailed Summary of Findings

Sl. No.	Name	Severity
M-01	Centralization Risk for trusted owners	Medium
I-01	Functions not used internally could be marked external	Informational
I-02	A proxy should be used to initialize the contract	Informational



FINDINGS

Static Analysis

No major issues were found. Some false positive errors were reported by the tool. All the other issues have been categorized above according to their level of severity.

Manual Review

High Severity Issues

No High Severity issues were found.

Medium Severity Issues

M-01. Centralization Risk for trusted owners

Description:

Contracts have owners with privileged rights to perform admin tasks and need to be trusted to not perform malicious updates or drain funds.

There exist 2 functions with **onlyOwner** access modifier. While the **snapshot** function may not lead to any problem due to being expected that only the owner can execute a snapshot, the **mint** function also implements the **onlyOwner** modifier. This **mint** function allow the owner to mint an unlimited amount of tokens for free.

FINDINGS

Code Snippets:

```
function snapshot() public onlyOwner {
    _snapshot();
}

ftrace | funcSig
function mint(address to, uint256 amount) public onlyOwner {
    _mint(to, amount);
}
```

Recommendations :

Do not allow the owner to mint an unlimited amount of tokens by free in case that it is not the expected behavior. If the expected behaviour is the mentioned one, clearly warn the users of the described fact as it could affect them.

Informational Issues

I-01. Functions not used internally could be marked external

Description:

There are some functions that are not used internally and are defined as **public** that can be defined as **external** instead.

FINDINGS

Code Snippets:

```
File: MyToken.sol

17:     function initialize() initializer public {

27:     function snapshot() public onlyOwner {

31:     function mint(address to, uint256 amount) public onlyOwner {
```

Recommendations :

Change the **public** declaration by **external**.

I-02. A proxy should be used to initialize the contract

Description:

The **initialize()** function may not work as expected.

Recommendations :

In order to work as expected the **initialize()** function should be called from a proxy using **delegatecall**.

POST AUDIT CONCLUSION

Fixing the Findings

Sl. No.	Name	Status
M-01	Centralization Risk for trusted owners	Acknowledged
I-01	Functions not used internally could be marked external	Acknowledged
I-02	A proxy should be used to initialize the contract	Acknowledged

Recommendations :

Fixing the issues provides a more clear code-base. All issues have been acknowledged by the project's developers.

APPENDIX

Finding Categories

Finding Categories

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.

Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM 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 how block.timestamp works.

Language Specific

Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of private or delete.

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.

DISCLAIMER

DetectBox (by UNSNARL) has conducted an independent security review to verify the integrity of and highlight any vulnerabilities, bugs or errors, intentional or unintentional, that may be present in the codebase that were provided for the scope of this Engagement.

Blockchain technology and the cryptographic assets it supports are nascent technologies. This makes them extremely volatile assets. Any assessment report obtained on such volatile and nascent assets may include unpredictable results which may lead to positive or negative outcomes.

In some cases, services provided may be reliant on a variety of third parties. This security review does not constitute endorsement, agreement or acceptance for the Project and technology that was reviewed. Users relying on this security review should not consider this as having any merit for financial advice or technological due diligence in any shape, form or nature.

The veracity and accuracy of the findings presented in this report relate solely to the proficiency, competence, aptitude and discretion of our auditors. UNSNARL and its employees make no guarantees, nor assurance that the contracts are free of exploits, bugs, vulnerabilities, deprecation of technologies or any system / economical / mathematical malfunction.

DISCLAIMER

All the information/opinions/suggestions provided in this report does not constitute financial or investment advice, nor should it be used to signal that any person reading this report should invest their funds without sufficient individual due diligence regardless of the findings presented in this report.

Information in this report is provided 'as is'. UNSNARL is under no covenant to the completeness, accuracy or solidity of the contracts reviewed. UNSNARL's goal is to help reduce the attack vectors/surface and the high level of variance associated with utilizing new and consistently changing technologies.

UNSNARL in no way claims any guarantee, warranty or assurance of security or functionality of the technology that was in scope for this security review.

In no event will UNSNARL, its partners, employees, agents or any parties related to the design/creation of this security review be ever liable to any parties for, or lack thereof, decisions and/or actions with regards to the information provided in this security review.

Cryptocurrencies and all other technologies directly or indirectly related to cryptocurrencies are not standardized, highly prone to malfunction and extremely speculative by nature. No due diligence and/or safeguards may be insufficient and users should exercise maximum caution when participating and/or investing in this nascent industry.

DISCLAIMER

The preparation of this security review has made all reasonable attempts to provide clear and actionable recommendations to the Project team (the “client”) with respect to the rectification, amendment and/or revision of any highlighted issues, vulnerabilities or exploits within the contracts in scope for this engagement.

It is the sole responsibility of the Project team to provide adequate levels of test and perform the necessary checks to ensure that the contracts are functioning as intended, and more specifically to ensure that the functions contained within the contracts in scope have the desired intended effects, functionalities and outcomes, as documented by the Project team.

All services, the security reports, discussions, work product, attack vectors description or any other materials, products or results of this security review engagement is provided "as is" and "as available" and with all faults, uncertainty and defects without warranty or guarantee of any kind.

UNSNARL will assume no liability or responsibility for delays, errors, mistakes, or any inaccuracies of content, suggestions, materials or for any loss, delay, damage of any kind which arose as a result of this engagement/security review.

UNSNARL will assume no liability or responsibility for any personal injury, property damage, of any kind whatsoever that resulted in this engagement and the customer having access to or use of the products, engineers, services, security report, or any other other materials.

For avoidance of doubt, this report, its content, access, and/or usage thereof, including any associated services or materials, shall not be considered or relied upon as any form of financial, investment, tax, legal, regulatory, or any other type of advice.

GLORY OF AUDITORS AT DETECTBOX

DetectBox has created a pool of best auditors across the globe with significant experience in the web3 security industry auditing multiple protocols across multiple chains. Making audits better through **Proof of Talent**.

217+

Projects Audited

$100M+

Amount Secured

2200+

Vulnerabilities Detected

Follow Our Journey on

@unsnarl_secure

UNSNARL

founders@unsnarl.io

www.detectbox.io