YEARN GENERIC AAVE SMART CONTRACT AUDIT

April 23, 2021

MixBytes()

CONTENTS

1. INTRODUCTION

1.1 DISCLAIMER

The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, investment advice, endorsement of the platform or its products, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bug free status. The audit documentation is for discussion purposes only. The information presented in this report is confidential and privileged. If you are reading this report, you agree to keep it confidential, not to copy, disclose or disseminate without the agreement of Yearn. If you are not the intended recipient(s) of this document, please note that any disclosure, copying or dissemination of its content is strictly forbidden.

1.2 PROJECT OVERVIEW

Part of Yearn Strategy Mix.

1.3 SECURITY ASSESSMENT METHODOLOGY

At least 2 auditors are involved in the work on the audit who check the provided source code independently of each other in accordance with the methodology described below:

01 "Blind" audit includes:
> Manual code study
> "Reverse" research and study of the architecture of the code based on the source code only

Stage goal:
Building an independent view of the project's architecture
Finding logical flaws

02 Checking the code against the checklist of known vulnerabilities includes:
> Manual code check for vulnerabilities from the company's internal checklist
> The company's checklist is constantly updated based on the analysis of hacks, research and audit of the clients' code

Stage goal:
Eliminate typical vulnerabilities (e.g. reentrancy, gas limit, flashloan attacks, etc.)

03 Checking the logic, architecture of the security model for compliance with the desired model, which includes:
> Detailed study of the project documentation
> Examining contracts tests
> Examining comments in code
> Comparison of the desired model obtained during the study with the reversed view obtained during the blind audit

Stage goal:
Detection of inconsistencies with the desired model

04 Consolidation of the reports from all auditors into one common interim report document
> Cross check: each auditor reviews the reports of the others
> Discussion of the found issues by the auditors
> Formation of a general (merged) report

Stage goal:
Re-check all the problems for relevance and correctness of the threat level
Provide the client with an interim report

05 Bug fixing & re-check.
> Client fixes or comments on every issue
> Upon completion of the bug fixing, the auditors double-check each fix and set the statuses with a link to the fix

Stage goal:
Preparation of the final code version with all the fixes

06 Preparation of the final audit report and delivery to the customer.

Findings discovered during the audit are classified as follows:

FINDINGS SEVERITY BREAKDOWN

Level	Description	Required action
Critical	Bugs leading to assets theft, fund access locking, or any other loss funds to be transferred to any party	Immediate action to fix issue
Major	Bugs that can trigger a contract failure. Further recovery is possible only by manual modification of the contract state or replacement.	Implement fix as soon as possible
Warning	Bugs that can break the intended contract logic or expose it to DoS attacks	Take into consideration and implement fix in certain period
Comment	Other issues and recommendations reported to/acknowledged by the team	Take into consideration

Based on the feedback received from the Customer's team regarding the list of findings discovered by the Contractor, they are assigned the following statuses:

Status	Description
Fixed	Recommended fixes have been made to the project code and no longer affect its security.
Acknowledged	The project team is aware of this finding. Recommendations for this finding are planned to be resolved in the future. This finding does not affect the overall safety of the project.
No issue	Finding does not affect the overall safety of the project and does not violate the logic of its work.

1.4 EXECUTIVE SUMMARY

The main purpose of the project is to give users ability to earning yield % with different lender schemes (GenericLender) managed by strategy. In our case the GenericAave scheme was scope.

1.5 PROJECT DASHBOARD

Client	Yearn
Audit name	GenericAave
Initial version	55b4d3b03845b7b71b24b50baa30823b3e42ebcf 7bd06e821732faa2b6d9f7da4b9d172f07649005
Final version	7bd06e821732faa2b6d9f7da4b9d172f07649005 7bd06e821732faa2b6d9f7da4b9d172f07649005
SLOC	131
Date	2021-04-09 - 2021-04-23
Auditors engaged	2 auditors

FILES LISTING

GenericAave.sol	GenericAave.sol

FINDINGS SUMMARY

Level	Amount
Critical	0
Major	1
Warning	1
Comment	0

CONCLUSION

Smart contracts have been audited and several suspicious places were found. During audit no critical issues were found, one issue was marked major as it might lead to undesired behavior. One issue was marked as warning. After working on audit report all issues were fixed or acknowledged by client.Final commit identifier with all fixes: `7bd06e821732faa2b6d9f7da4b9d172f07649005`, `7bd06e821732faa2b6d9f7da4b9d172f07649005`

2. FINDINGS REPORT

2.1 CRITICAL

Not Found

2.2 MAJOR

MJR-1	Deposit will be unavailable if lending pool address will be updated by AAVE
File	GenericAave.sol
Severity	Major
Status	Fixed at PR-11

DESCRIPTION

At line GenericAave.sol#L132 the `deposit` function assumes recent approval of token transfer. However, the `safeApprove()` is called once during contract initialization(GenericAave.sol#L49) and possible changes of lending pool address is not tracked properly. If lending pool address is updated by AAVE, the `deposit()` will be unavailable/reverted until contract replacement.

RECOMMENDATION

Call `safeApprove()` on demand before calling `deposit()` on lending pool.

2.3 WARNING

WRN-1	The approval value obtained in the `_initialize()` function may not be enough for the long term of the smart contract
File	GenericAave.sol
Severity	Warning
Status	Acknowledged

DESCRIPTION

Smart contracts call `safeApprove()` function for different tokens. But in the process of work, the obtained value will only decrease. If this value decreases to zero, then the tokens will remain locked in the contract forever.
It is at the following lines:

- GenericAave.sol#L49

RECOMMENDATION

It is recommended to add a function to increase the value of approvals.

2.4 COMMENTS

Not Found

3. ABOUT MIXBYTES

MixBytes is a team of blockchain developers, auditors and analysts keen on decentralized systems. We build open-source solutions, smart contracts and blockchain protocols, perform security audits, work on benchmarking and software testing solutions, do research and tech consultancy.

BLOCKCHAINS

 Ethereum

 Cosmos

 EOS

 Substrate

TECH STACK

 Python

 Solidity

 Rust

 C++

CONTACTS

 https://github.com/mixbytes/audits_public

 https://mixbytes.io/

 hello@mixbytes.io

 https://t.me/MixBytes

 https://twitter.com/mixbytes