B-CUBE ICO SMART CONTRACT AUDIT

February 15, 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 B-Cube . 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

B-Cube is a marketplace of AI-driven crypto trading bots which allows traders connecting to their favorite exchanges and start trading on auto-pilot.

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 checked volume includes functionality that performs a private crowdsale of BCUBE tokens to private investors, accepting payments in $ETH and $USDT, using Chainlink's price feed to offer $BCUBE tokens to investors. In addition to private investors, the team and advisors will receive their share of allocated BCUBE tokens in a vested manner.

1.5 PROJECT DASHBOARD

Client	B-Cube
Audit name	ICO
Initial version	451e249a7200ea094fdfa1baa1a50cb7b17233f2
Final version	09efaa97fa92f6a4e31b10cd1d93b2b4e80eba31
SLOC	554
Date	2021-02-02 - 2021-02-15
Auditors engaged	2 auditors

FILES LISTING

BCubePrivateSale.sol	BCubePrivateSale.sol
Treasury.sol	Treasury.sol
Staking.sol	Staking.sol
BCUBEToken.sol	BCUBEToken.sol

FINDINGS SUMMARY

Level	Amount
Critical	0
Major	0
Warning	3
Comment	0

CONCLUSION

Findings list: Level Amount CRITICAL 0 MAJOR 0 WARNING 3 COMMENT 0 Final commit identifier with all fixes: `09efaa97fa92f6a4e31b10cd1d93b2b4e80eba31`

2. FINDINGS REPORT

2.1 CRITICAL

Not Found

2.2 MAJOR

Not Found

2.3 WARNING

WRN-1	Potentially unrestricted account withdrawals
File	Treasury.sol
Severity	Warning
Status	Fixed at 09efaa97

DESCRIPTION

This warning is about any user potentially being able to withdraw advisor's share in here: Treasury.sol#L158 and private sale participant's share in here: Treasury.sol#L308.

Some check if `msg.sender` is an advisor is being performed in here: Treasury.sol#L163 along with a check if `msg.sender` is a private sale participant in here: Treasury.sol#L163, but it could be done better.

RECOMMENDATION

It is recommended to introduce a more explicit advisor and private sale membership check with a function modifier just like it was done with OpenZeppelin's `onlyWhitelistAdmin`: https://docs.openzeppelin.com/contracts/2.x/api/access#WhitelistAdminRole-onlyWhitelistAdmin--.

WRN-2	Potentially incorrect allowance amount
File	Treasury.sol
Severity	Warning
Status	Fixed at 09efaa97

DESCRIPTION

This warning concerns the potentially incorrect computation of the allowance for advisors and private sale members in here:

- Treasury.sol#L165
- Treasury.sol#L166
- Treasury.sol#L167.

RECOMMENDATION

It is recommended to introduce `SafeMath` usage for calculating allowance `increase` values.

WRN-3	Potentially incorrect price data
File	BCubePrivateSale.sol
Severity	Warning
Status	Fixed at 09efaa97

DESCRIPTION

This warning is about potentially incorrect price data being calculated from Chainlink oracle results in here: BCubePrivateSale.sol#L112. Chainlink output can be of a little bit more complicated format than it is expected in the most trivial case (e.g. https://blog.chain.link/fetch-current-crypto-price-data-solidity/).
At the line BCubePrivateSale.sol#L120, arithmetic operations are performed on the query results without using the `SafeMath`.

RECOMMENDATION

It is recommended to pay attention to handling Chainlink output data format correctly (in case it is not yet) and handle all the arithmetic operations with it with `SafeMath` usage.

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