STAYSAFU
AUDIT

APRIL 13TH, 2022

BATCOIN

TABLE OF CONTENTS

AUDIT SUMMARY

This report was written for Batcoin(BATS) in order to find flaws and vulnerabilities in the Batcoin project's source code, as well as any contract dependencies that weren't part of an officially recognized library.

A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and Batcoin Deployment 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
- Through line-by-line manual review of the entire codebase by industry expert

AUDIT OVERVIEW

PROJECT SUMMARY

Project name	Batcoin
Description	Batcoin is a community driven token that will allow users to give experiences about metaverse, game, swap and blockchain communication projects
Platform	Binance Smart Chain
Language	Solidity
Codebase	https://bscscan.com/token/0xaa731eb48f701c132d5fa69b69f5eb4064d9be34

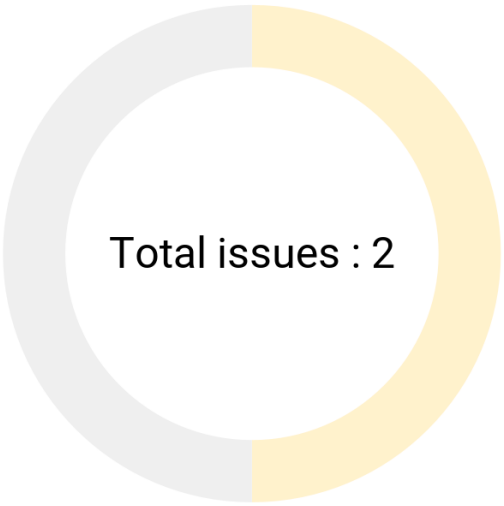
FINDINGS SUMMARY

Vulnerability	Total
Critical	0
Major	0
Medium	0
Minor	1
Informational	1

EXECUTIVE SUMMARY

Batcoin is a digital currency that will create metaverse, game, swap and blockchain communication projects. Batcoin also aims to be of good value to help the animal foundations and preservation organizations. The developer deployed a safe contract that has only 2% tax reflection with locked liquidity, allowing it to grow organically. Batcoin is a purely community driven coin and aims to be one of the top leading coins in the crypto industry.

AUDIT FINDINGS



Code	Title	Severity
COMP-1	Unfixed version of compiler	Minor
MSG-3	Missing error message	Informational

MSG-3 | Too long error message

Description

The smart contract has some error messages that are too long. The industry standards specify error messages must have a maximal length of 32 bytes. We recommend having the shortest possible error messages to optimize gas costs (see github.com/ethereum/solidity/issues/4588) and improve error handling. 10 issues of this type have been found in the smart contract.

Recommendation

We recommend replacing these ambiguous messages with clearer ones :

```
//Edited code with clearer error messages
//l287
require(success, "Address : unable to send value");
//l488
_approve(sender, _msgSender(),
_allowances[sender][_msgSender()].sub(amount, "transfer
amount over allowance"));
//l498
_approve(_msgSender(), spender,
_allowances[_msgSender()][spender].sub(subtractedValue,
"ERC20: allowance under 0"));
//l512
require(!_isExcluded[sender], "Excluded addresses");
//l531
require(rAmount <= _rTotal, "Amount over total
```

```
reflections");
//l559
require(owner != address(0), "ERC20 approve from the 0
address");
//l560
require(owner != address(0), "ERC20 approve from the 0
address");
//l567
require(sender != address(0), "Transfer from the zero
address");
//l568
require(sender != address(0), "Transfer from the zero
address");
//l569
require(amount > 0, "Amount lower than 0");
```

COMP-1 | Unfixed version of compiler

Description

Batcoin's contract does not have locked compiler versions, meaning a range of compiler versions can be used. This can lead to differing bytecodes being produced depending on the compiler version, which can create confusion when debugging as bugs may be specific to a specific compiler version(s).

To rectify this, we recommend setting the compiler to a single version, the lowest version tested to be compatible with the code.

1 error of this type has been found in the smart contract.

Recommendation

We recommend fixing the compiler version to the most recent one :

```
//Edited code containing fixed compiler version
//l10
pragma solidity 0.6.12;
```

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 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 without StaySAFU's prior written consent.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 StaySAFU 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.

StaySAFU's position is that each company and individual are responsible for their own due diligence and continuous security. StaySAFU'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 fun.