AuditAce
FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

VERB

DATED : 26 Dec 23'

AUDIT SUMMARY

Project name – VerbChain

Date: 26 Dec, 2023

Scope of Audit- Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

Audit Status: Passed

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	0	1	1	0
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0

USED TOOLS

Tools:

1- Manual Review:
A line by line code review has been performed by audit ace team.

2- BSC Test Network: All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it.

3- Slither :
The code has undergone static analysis using Slither.

Testnet version:

The tests were performed using the contract deployed on the BSC, which can be found at the following address:
https://bscscan.com/address/0x87eC0CC13b739b2443C9A19CD8aa7a576da9621a#code

Token Information

Token Address:
0x87eC0CC13b739b2443C9A19CD8aa7a576da9621a

Name: VERB

Symbol: VERB

Decimals: 18

Network: BscScan

Token Type: BEP-20

Owner:
0x9d487a34bb27a2a359b4eb41df22cc7f0a951455

Deployer:
0x9d487a34bb27a2a359b4eb41df22cc7f0a951455

Token Supply: 100000000000000000000000000

Mainnet:
https://bscscan.com/address/0x87eC0CC13b739b2443C9A19CD8aa7a576da9621a

TOKEN OVERVIEW

Buy Fee: 0-1%

Sell Fee: 0-1%

Transfer Fee: 0-0%

Fee Privilege: Owner

Ownership: Owned

Minting: None

Max Tx: Yes

Blacklist: No

AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.
- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.
- Test coverage analysis determines whether the test cases are covering the code and how much code isexercised when we run the test cases.
- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.
- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

VULNERABILITY CHECKLIST

- Return values of low-level calls
- Private modifier
- Multiple Sends
- Using Suicide
- Gas Limitand Loops
- Address hardcoded
- Exception Disorder
- Using inline assembly
- Divide before multiply
- Missing Zero Address Validation
- Compiler version not fixed
- **Gasless Send**
- Using block.timestamp
- Re-entrancy
- Tautology or contradiction
- Timestamp Dependence
- Revert/require functions
- Use of tx.origin
- Integer overflow/underflow
- Dangerous strict equalities
- Using SHA3
- Using throw

CLASSIFICATION OF RISK

Severity	Description
◆ Critical	These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
◆ High-Risk	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ Medium-Risk	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ Low-Risk	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ Gas Optimization Suggestion	A vulnerability that has an informational character but is not affecting any of the code.

Findings

Severity	Found
◆ Critical	0
◆ High-Risk	0
◆ Medium-Risk	0
◆ Low-Risk	0
◆ Gas Optimization Suggestions	0

POINTS TO NOTE

- The owner can renounce the ownership.
- The owner can transfer ownership.
- The owner can change the marketing wallet address.
- The owner can Include/exclude the address from fees.

STATIC ANALYSIS

Result => A static analysis of contract's source code has been performed using slither,

No major issues were found in the output

MANUAL TESTING

Centralization – Liquidity is added to EOA. Severity: Medium
function: _addLiquidity
Status: Open

Overview:
Liquidity is added to EOA. It may be drained by the addLiquidityETH.

```
function _addLiquidity(uint256 tokenAmount, uint256 ethAmount)
private
lockTheSwap
 {
_approve(address(this), address(uniswapV2Router),
tokenAmount);

uniswapV2Router.addLiquidityETH{value: ethAmount}(
address(this),
tokenAmount,
0,
0,
marketingWallet,
block.timestamp
);
 }
```

Suggestion:
It is suggested that the address should be a contract address or a dead address.

MANUAL TESTING

Centralization – Missing Events Severity: Low

subject: Missing Events

Status: Open

Overview:
They serve as a mechanism for emitting and recording data onto the blockchain, making it transparent and easily accessible.

```
 function changeMarketingWallet(address newWallet)
public
 onlyOwner
 returns (bool)
  {
 marketingWallet = newWallet;
 return true;
 function changeTaxForLiquidityAndMarketing(uint256
_taxBuyForLiquidity, uint256 _taxBuyForMarketing, uint256
_taxSellForLiquidity, uint256 _taxSellForMarketing)
 public
 onlyOwner
 returns (bool)
  {
 require((_taxBuyForLiquidity+_taxBuyForMarketing) <= 20,
"ERC20: total tax must not be greater than 20");
 require((_taxSellForLiquidity+_taxSellForMarketing) <= 20,
"ERC20: total tax must not be greater than 20");
 taxBuyForLiquidity = _taxBuyForLiquidity;
 taxBuyForMarketing = _taxBuyForMarketing;
```

MANUAL TESTING

```
taxSellForLiquidity = _taxSellForLiquidity;
taxSellForMarketing = _taxSellForMarketing;

return true;
}

function changeMaxTxAmount(uint256 _maxTxAmount)
public
onlyOwner
returns (bool)
{
require(_maxTxAmount >= 100_000_000, "ERC20: maxTxAmount
must not be less than 0.1% total supply");
maxTxAmount = _maxTxAmount * 10**_decimals;

return true;
}
function changeMaxWalletAmount(uint256 _maxWalletAmount)
public
onlyOwner
returns (bool)
{
require(_maxWalletAmount >= 100_000_000, "ERC20:
maxWalletAmount must not be less than 0.1% total supply");
maxWalletAmount = _maxWalletAmount * 10**_decimals;

return true;
}
```

DISCLAIMER

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment. Team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token. The Auditace team disclaims any liability for the resulting losses.

ABOUT AUDITACE

We specializes in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.

https://auditace.tech/

https://t.me/Audit_Ace

https://twitter.com/auditace_