AuditAce
FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

Xmonsters

DATED : 08 Jan, 2024

MANUAL TESTING

Centralization – Buy and Sell Fees.

Severity: High
Function: changeTaxForMarketing
Status: Open

Overview:
The owner can set the buy and sell fees to more than 30%, which is not recommended.

```
function changeTaxForMarketing(
uint256 _taxBuy,
uint256 _taxSell
 ) public onlyOwner returns (bool) {
require(
    (_taxBuy + _taxSell) <= 30,
"ERC20: total tax must not be greater than 30"
   );
   _taxBuyForMarketing = _taxBuy;
   _taxSellForMarketing = _taxSell;
return true;
 }
```

Suggestion
It is recommended that no fees in the contract should be more than 25% of the contract.

AUDIT SUMMARY

Project name – Xmonsters

Date: 08 Jan, 2024

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

Audit Status: **Passed with High Risk**

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	1	1	1	1
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. These tests can be found in the "Functional Tests" section of the report.

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

Testnet version:
The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:
https://testnet.bscscan.com/address/0xc67957d330228c380f0e77a5fec9519824a70a2c#code

Token Information

Token Address:
0xC6237E7173B726dB057D8ecd7Aa10f6d718144a3

Name: Xmonsters

Symbol: XMON

Decimals: 9

Network: BscScan

Token Type: BEP-20

Owner:
0xeeAc6c4EBb0EDc7C6DF680718bc100745BF30e28

Deployer:
0xeeAc6c4EBb0EDc7C6DF680718bc100745BF30e28

Token Supply: 10,000,000

Checksum: A67acbefe2a12642d388659dffd207fc

Testnet:
https://testnet.bscscan.com/address/0xc67957d330228c380f0e77a5fec9519824a70a2c#code

TOKEN OVERVIEW

Buy Fee: 3-30%

Sell Fee: 3-30%

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

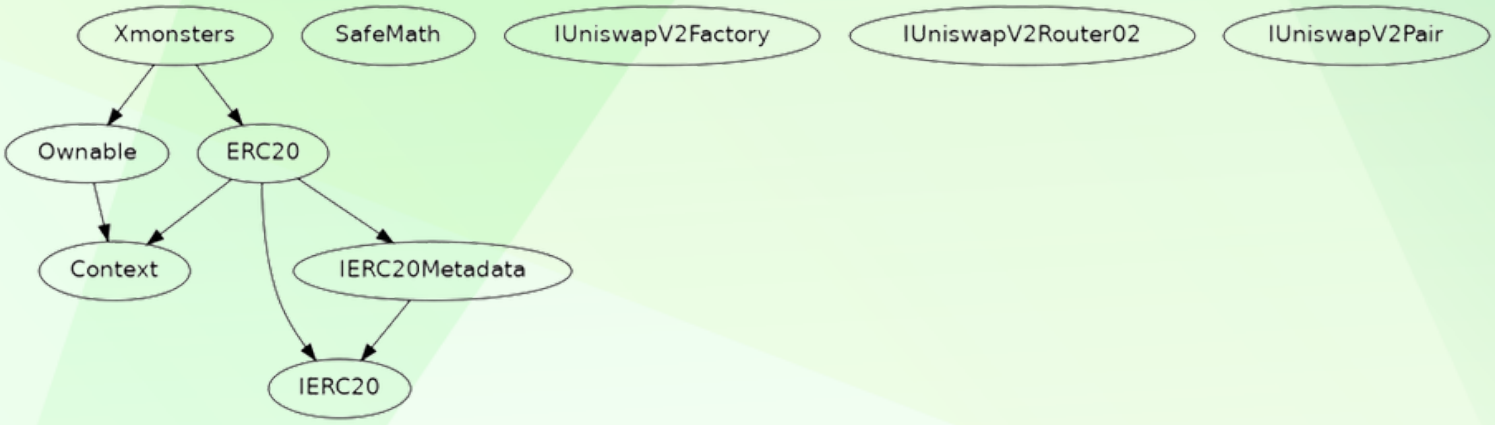
CLASSIFICATION OF RISK

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	1
Medium-Risk	1
Low-Risk	1
Gas Optimization / Suggestions	1

INHERITANCE TREE



POINTS TO NOTE

- The owner can renounce ownership.
- The owner can change the buy and sell tax by more than 25%.
- The owner can change the marketing wallet address.

STATIC ANALYSIS

```
INFO:Detectors:
Contract locking ether found:
        Contract Xmonsters (Xmonsters.sol#654-769) has payable functions:
         - Xmonsters.receive() (Xmonsters.sol#767)
        But does not have a function to withdraw the ether
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#contracts-that-lock-ether
INFO:Detectors:
Xmonsters.constructor().currentRouter (Xmonsters.sol#682) is a local variable never initialized
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#uninitialized-local-variables
INFO:Detectors:
Xmonsters.changeTaxForMarketing(uint256,uint256) (Xmonsters.sol#754-765) should emit an event for:
        - _taxBuyForMarketing = _taxBuy (Xmonsters.sol#762)
        - _taxSellForMarketing = _taxSell (Xmonsters.sol#763)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-arithmetic
INFO:Detectors:
Xmonsters.changeMarketingWallet(address).newWallet (Xmonsters.sol#738) lacks a zero-check on :
                - marketingWallet = newWallet (Xmonsters.sol#740)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
INFO:Detectors:
ERC20._burn(address,uint256) (Xmonsters.sol#566-578) is never used and should be removed
SafeMath.add(uint256,uint256) (Xmonsters.sol#119-123) is never used and should be removed
SafeMath.div(uint256,uint256) (Xmonsters.sol#148-150) is never used and should be removed
SafeMath.div(uint256,uint256,string) (Xmonsters.sol#152-160) is never used and should be removed
SafeMath.mul(uint256,uint256) (Xmonsters.sol#139-146) is never used and should be removed
SafeMath.sub(uint256,uint256) (Xmonsters.sol#125-127) is never used and should be removed
SafeMath.sub(uint256,uint256,string) (Xmonsters.sol#129-137) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version0.8.19 (Xmonsters.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.8.18.
solc-0.8.19 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Function IUniswapV2Router02.WETH() (Xmonsters.sol#209) is not in mixedCase
Function IUniswapV2Pair.DOMAIN_SEPARATOR() (Xmonsters.sol#253) is not in mixedCase
Function IUniswapV2Pair.PERMIT_TYPEHASH() (Xmonsters.sol#255) is not in mixedCase
Function IUniswapV2Pair.MINIMUM_LIQUIDITY() (Xmonsters.sol#286) is not in mixedCase
Parameter Xmonsters.setIsExcludeFee(address[],bool)._address (Xmonsters.sol#745) is not in mixedCase
Parameter Xmonsters.changeTaxForMarketing(uint256,uint256)._taxBuy (Xmonsters.sol#755) is not in mixedCase
```

```
INFO:Detectors:
ERC20._burn(address,uint256) (Xmonsters.sol#566-578) is never used and should be removed
SafeMath.add(uint256,uint256) (Xmonsters.sol#119-123) is never used and should be removed
SafeMath.div(uint256,uint256) (Xmonsters.sol#148-150) is never used and should be removed
SafeMath.div(uint256,uint256,string) (Xmonsters.sol#152-160) is never used and should be removed
SafeMath.mul(uint256,uint256) (Xmonsters.sol#139-146) is never used and should be removed
SafeMath.sub(uint256,uint256) (Xmonsters.sol#125-127) is never used and should be removed
SafeMath.sub(uint256,uint256,string) (Xmonsters.sol#129-137) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version0.8.19 (Xmonsters.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.8.18.
solc-0.8.19 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Function IUniswapV2Router02.WETH() (Xmonsters.sol#209) is not in mixedCase
Function IUniswapV2Pair.DOMAIN_SEPARATOR() (Xmonsters.sol#253) is not in mixedCase
Function IUniswapV2Pair.PERMIT_TYPEHASH() (Xmonsters.sol#255) is not in mixedCase
Function IUniswapV2Pair.MINIMUM_LIQUIDITY() (Xmonsters.sol#286) is not in mixedCase
Parameter Xmonsters.setIsExcludeFee(address[],bool)._address (Xmonsters.sol#745) is not in mixedCase
Parameter Xmonsters.changeTaxForMarketing(uint256,uint256)._taxBuy (Xmonsters.sol#755) is not in mixedCase
Parameter Xmonsters.changeTaxForMarketing(uint256,uint256)._taxSell (Xmonsters.sol#756) is not in mixedCase
Variable Xmonsters._taxBuyForMarketing (Xmonsters.sol#660) is not in mixedCase
Variable Xmonsters._taxSellForMarketing (Xmonsters.sol#661) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
Xmonsters._decimals (Xmonsters.sol#658) should be constant
Xmonsters._name (Xmonsters.sol#656) should be constant
Xmonsters._supply (Xmonsters.sol#659) should be constant
Xmonsters._symbol (Xmonsters.sol#657) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
INFO:Slither:Xmonsters.sol analyzed (10 contracts with 93 detectors), 28 result(s) found
```

FUNCTIONAL TESTING

1- **Approve** (passed):
https://testnet.bscscan.com/tx/0x28046d3599450fe14d7ef72290935f174bc0ad64ddcb82d79ab359089b32ee6e

2- **Increase Allowance** (passed):
https://testnet.bscscan.com/tx/0xb331f3d24753f8c02f7dfc6962956a1f6d69425f8054b4d881773001f99bd4ff

3- **Decrease Allowance** (passed):
https://testnet.bscscan.com/tx/0x3b6b9920c90ca9b1eaed4b55d72550bbca4ee9f6ceaf42783eaf6cb0e293af0b

4- **Change Tax for Marketing** (passed):
https://testnet.bscscan.com/tx/0x6c0fa87b4f37dfe76c30be4e9ab30f2507669e4e9b1eba784ae2c5c527accc72

MANUAL TESTING

Centralization – Buy and Sell Fees.

Severity: High
Function: changeTaxForMarketing
Status: Open

Overview:
The owner can set the buy and sell fees to more than 30%, which is not recommended.

```
function changeTaxForMarketing(
uint256 _taxBuy,
uint256 _taxSell
 ) public onlyOwner returns (bool) {
require(
    (_taxBuy + _taxSell) <= 30,
"ERC20: total tax must not be greater than 30"
   );
   _taxBuyForMarketing = _taxBuy;
   _taxSellForMarketing = _taxSell;
return true;
 }
```

Suggestion
It is recommended that no fees in the contract should be more than 25% of the contract.

MANUAL TESTING

Centralization –Missing Require Check.
Severity: Medium
Function: changeMarketingWallet
Status: Open

Overview:
The owner can set any arbitrary address excluding zero address as this is not recommended because if the owner will set the address to the contract address, then the Eth will not be sent to that address and the transaction will fail and this will lead to a potential honeypot in the contract.

```
function changeMarketingWallet(
address newWallet
  ) public onlyOwner returns (bool) {
    marketingWallet = newWallet;
return true;
  }
```

Suggestion:
It is recommended that the address should not be able to be set as a contract 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 changeTaxForMarketing(
uint256 _taxBuy,
uint256 _taxSell
  ) public onlyOwner returns (bool) {
require(
     (_taxBuy + _taxSell) <= 30,
"ERC20: total tax must not be greater than 30"
    );
    _taxBuyForMarketing = _taxBuy;
    _taxSellForMarketing = _taxSell;
return true;
  }
```

MANUAL TESTING

```
function changeMarketingWallet(
address newWallet
 ) public onlyOwner returns (bool) {
 marketingWallet = newWallet;
return true;
 function setIsExcludeFee(
address[] memory _address,
bool state
 ) public onlyOwner returns (bool) {
for (uint256 i = 0; i < _address.length; i++) {
 _isExcludedFromFee[_address[i]] = state;
 }
return true;
 }
```

MANUAL TESTING

Severity: Informational
Subject: Remove Safe Math
Status: Open
Line: 118-161

Overview:
compiler version above 0.8.0 has the ability to control arithmetic overflow/underflow, It is recommended to remove the unwanted code in order to avoid high gas fees.

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_

https://github.com/Audit-Ace