SAFUAUDIT
SMART CONTRACT AUDITING

GOLDMINER

SMART CONTRACT AUDIT

March 14, 2022

INTRODUCTION

Client	GoldMiner (GOLD)
Language	Solidity
Contract address	0x237E36ADdBe39a2997966AFF8780ea876f03986d
Decimals	18
Supply	100,000,000
Platform	Binance Smart Chain
Compiler	v0.8.7+commit.e28d00a7
Optimization	Yes, with 200 runs
Website	https://goldminers.io
Telegram	https://t.me/GoldMinerchat
Twitter	https://twitter.com/GoldminersBsc

Description

GoldMiners is a Play-to-Earn game on the Binance Smart Chain that involves generating the best possible computing power. The goal is to win GOLD by fighting enemies with computing power as your main weapon. You will have to make strategic decisions to achieve your goal.

TABLE OF CONTENTS

01 INTRODUCTION

Introduction 02

Approach 04

Risk classification 05

02 ABSTRACT

Abstract 06

03 VULNERABILITIES TEST

Vulnerabilities Test 07

04 MANUAL ANALYSIS

Manual analysis 09

Contract Inspection 10

Inheritance Tree 13

Important Snippets 14

Good Practices 15

05 WEBSITE

Website Audit 16

06 CONCLUSIONS

Disclaimer 17

Audit Results 18

SafuScore 19

Summary 20

Approach

Audit Details

Our comprehensive audit report provides a full overview of the audited system's architecture, smart contract codebase, and details on any vulnerabilities found within the system.

Audit Goals

The audit goal is to ensure that the project is built to protect investors and users, preventing potentially catastrophic vulnerabilities after launch, that lead to scams and rugpulls.

,

Code Quality

Our analysis includes both automatic tests and manual code analysis for the following aspects:

- Exploits
- Back-doors
- Vulnerability
- Accuracy
- Readability

Tools

- Remix IDE
- MythX, Mythril
- SWC Registry
- Open Zeppelin Code Analyzer
- Solidity Code Complier

RISK CLASSIFICATION

CRITICAL

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

MEDIUM

Issues on this level could potentially bring problems and should eventually be fixed.

MINOR

Issues on this level are minor details and warning that can remain unfixed but would be better fixed at some point in the future

INFORMATIONAL

Information level is to offer suggestions for improvement of efficacity or security for features with a risk free factor.

ABSTRACT

Fees	Ownership	Mint function
Buy Fees: 0% Sell Fees: 5%	Owned	No mint function found

Owner can set fees	Max Tx amount	Pause
Owner can't set fees over 15%	Owner can't set max tx amount	Owner can't pause trading

Vulnerabilities Test

SWC ID	Description	
SWC-100	Function Default Visibility	Passed
SWC-101	Integer Overflow and Underflow	Passed
SWC-102	Outdated Compiler Version	Passed
SWC-103	Floating Pragma	Passed
SWC-104	Unchecked Call Return Value	Passed
SWC-105	Unprotected Ether Withdrawal	Passed
SWC-106	Unprotected SELF-DESTRUCT Instruction	Passed
SWC-107	Re-entrancy	Passed
SWC-108	State Variable Default Visibility	Passed
SWC-109	Uninitialized Storage Pointer	Passed
SWC-110	Assert Violation	Passed
SWC-111	Use of Deprecated Solidity Functions	Passed
SWC-112	Delegate Call to Untrusted Callee	Passed
SWC-113	DoS with Failed Call	Passed
SWC-114	Transaction Order Dependence	Passed
SWC-115	Authorization through tx.origin	Passed

SWC-116	Block values as a proxy for time	**Passed**
SWC-117	Signature Malleability	**Passed**
SWC-118	Incorrect Constructor Name	**Passed**
SWC-119	Shadowing State Variables	**Passed**
SWC-120	Weak Sources of Randomness from Chain Attributes	**Passed**
SWC-121	Missing Protection against Signature Replay Attacks	**Passed**
SWC-122	Lack of Proper Signature Verification	**Passed**
SWC-123	Requirement Violation	**Passed**
SWC-124	Write to Arbitrary Storage Location	**Passed**
SWC-125	Incorrect Inheritance Order	**Passed**
SWC-126	Insufficient Gas Griefing	**Passed**
SWC-127	Arbitrary Jump with Function Type Variable	**Passed**
SWC-128	DoS With Block Gas Limit	**Passed**
SWC-129	Typographical Error	**Passed**
SWC-130	Right-To-Left-Override control character (U+202E)	**Passed**
SWC-131	Presence of unused variables	**Passed**
SWC-132	Unexpected Ether balance	**Passed**
SWC-133	Hash Collisions With Multiple Variable Length Arguments	**Passed**
SWC-134	Message call with the hardcoded gas amount	**Passed**
SWC-135	Code With No Effects (Irrelevant/Dead Code)	**Passed**
SWC-136	Unencrypted Private Data On-Chain	**Passed**

MANUAL ANALYSIS

The contract is verified to check if functions do and work as they should and malicious code is not inserted.

	Tested	Result
Transfer	Yes	**Passed**
Total Supply	Yes	**Passed**
Buy Back	Yes	**N/A**
Burn	Yes	**N/A**
Mint	Yes	**N/A**
Rebase	Yes	**N/A**
Pause	Yes	**N/A**
Blacklist	Yes	**N/A**
Lock	Yes	**N/A**
Max Transaction	Yes	**N/A**
Transfer Ownership	Yes	**Passed**
Renounce Ownership	Yes	**Passed**

CONTRACT INSPECTION

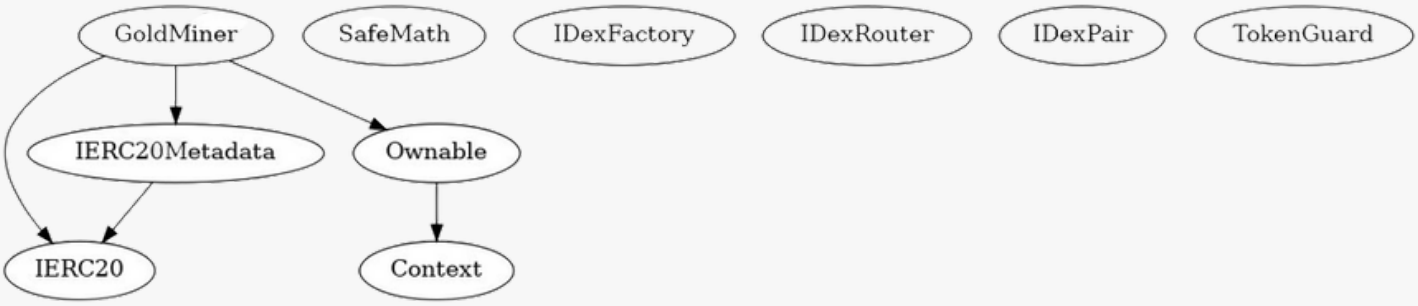
IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
IERC20Metadata	Interface	IERC20		
└	name	External ❗		NO❗
└	symbol	External ❗		NO❗
└	decimals	External ❗		NO❗
SafeMath	Library			
└	tryAdd	Internal 🔒		
└	trySub	Internal 🔒		
└	tryMul	Internal 🔒		
└	tryDiv	Internal 🔒		
└	tryMod	Internal 🔒		
└	add	Internal 🔒		
└	sub	Internal 🔒		
└	mul	Internal 🔒		
└	div	Internal 🔒		
└	mod	Internal 🔒		
└	sub	Internal 🔒		
└	div	Internal 🔒		
└	mod	Internal 🔒		
Context	Implementation			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
Ownable	Implementation	Context		
└	<Constructor>	Public ❗	🛑	NO❗
└	owner	Public ❗		NO❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
└	_setOwner	Internal 🔒	🛑	

IDexFactory	Interface			
└	createPair	External ❗	🛑	NO❗
└	pairFor	External ❗		NO❗
└	getPair	External ❗		NO❗
IDexRouter	Interface			
└	WETH	External ❗		NO❗
└	factory	External ❗		NO❗
└	swapExactTokensForETH	External ❗	🛑	NO❗
└	swapExactTokensForTokens	External ❗	🛑	NO❗
IDexPair	Interface			
└	token0	External ❗		NO❗
└	token1	External ❗		NO❗
└	getReserves	External ❗		NO❗
TokenGuard	Interface			
└	protect	External ❗	🛑	NO❗
GoldMiner	Implementation	IERC20, IERC20Metadata, Ownable		
└	<Constructor>	Public ❗	🛑	NO❗
└	changeWithdrawAddress	External ❗	🛑	onlyOwner
└	setSupportedDex	External ❗	🛑	onlyOwner
└	setDefaultSwapRouter	External ❗	🛑	onlyOwner
└	_setSupportedDex	Internal 🔒	🛑	
└	removeSupportedDex	External ❗	🛑	onlyOwner
└	amountForEth	Public ❗		NO❗
└	setTGAddress	External ❗	🛑	onlyOwner
└	excludeFromFee	External ❗	🛑	onlyOwner
└	includeInFee	External ❗	🛑	onlyOwner
└	setTaxAddress	External ❗	🛑	onlyOwner
└	setTax	External ❗	🛑	onlyOwner
└	isExcludedFromFee	External ❗		NO❗
└	withdrawExternalToken	External ❗	🛑	onlyOwner
└	withdrawEth	External ❗	🛑	onlyOwner
└	name	External ❗		NO❗
└	symbol	External ❗		NO❗
└	decimals	External ❗		NO❗

└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
└	increaseAllowance	External ❗	🛑	NO❗
└	decreaseAllowance	External ❗	🛑	NO❗
└	swapTokensForEth	Private 🔐	🛑	
└	swapTokensForToken	Private 🔐	🛑	
└	_transfer	Internal 🔒	🛑	
└	_approve	Internal 🔒	🛑	

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable
🔐	Private function
🔒	Internal function
NO❗	Function has no modifier

INHERITANCE TREE



Inheritance is a feature of the object-oriented programming language. It is a way of extending the functionality of a program, used to separate the code, reduces the dependency, and increases the re-usability of the existing code. Solidity supports inheritance between smart contracts, where multiple contracts can be inherited into a single contract.

Important Snippets

Owner can exclude from fees

```
function excludeFromFee(address account) external onlyOwner{
    _isExcludedFromFee[account] = true;
}

function includeInFee(address account) external onlyOwner{
    _isExcludedFromFee[account] = false;
}
```

Owner can only set sell fees up to 15%

```
function setTax(uint256 _sellFee) external onlyOwner{
    require(_sellFee<=1500,"invalid taxFee");
    sellTaxFee = _sellFee;
}
```

GOOD PRACTICES

- The owner cannot set maxTx
- The owner cannot set the fees over 15%
- The owner cannot stop or pause the smart contract
- The owner cannot mint tokens after initial deployment
- The smart contract utilizes "SafeMath" to prevent overflows

```
library SafeMath {
    function tryAdd(uint256 a, uint256 b) internal pure returns (bool, uint2  ,) {
        unchecked {
            uint256 c = a + b;
            if (c < a) return (false, 0);
            return (true, c);
        }
    }

    function trySub(uint256 a, uint256 b) internal pure returns (bool, uint256) {
        unchecked {
            if (b > a) return (false, 0);
            return (true, a - b);
        }
    }

    function tryMul(uint256 a, uint256 b) internal pure returns (bool, uint256) {
        unchecked {
            // Gas optimization: this is cheaper than requiring 'a' not being zero, but
            // benefit is lost if 'b' is also tested.
            // See: https://github.com/OpenZeppelin/openzeppelin-contracts/pull/522
            if (a == 0) return (true, 0);
            uint256 c = a * b;
            if (c / a != b) return (false, 0);
            return (true, c);
        }
    }

    function tryDiv(uint256 a, uint256 b) internal pure returns (bool, uint256) {
        unchecked {
            if (b == 0) return (false, 0);
            return (true, a / b);
        }
    }

    function tryMod(uint256 a, uint256 b) internal pure returns (bool, uint256) {
        unchecked {
            if (b == 0) return (false, 0);
            return (true, a % b);
        }
    }
```

WEBSITE

Website	https://goldminers.io/
Domain Registry	https://www.namecheap.com/
Domain Expiry Date	2024-01-15
Response Code	200
SSL Checker and HTTPS Test	Passed
Deprecated HTML tags	Passed
Robots.txt	Informational
Sitemap Test	Informational
SEO Friendly URL	Passed
Responsive Test	Passed
JS Error Test	Passed
Console Errors Test	Passed
Site Loading Speed Test	0.99 seconds - Passed
HTTP2 Test	Passed
Safe Browsing Test	Passed

DISCLAIMER

SafuAudit.com is not a financial institution and the information provided on this website does not constitute investment advice, financial advice, trading advice or any other sort of advice. You should not treat any of the website's content as such. Investing in crypto assets carries a high level of risk and does not hold guarantees for not sustaining financial loss due to their volatility.

Accuracy of Information

SafuAudit will strive to ensure accuracy of information listed on this website although it will not hold any responsibility for any missing or wrong information. SafuAudit provides all information as is. You understand that you are using any and all information available here at your own risk. Any use or reliance on our content and services is solely at your own risk and discretion.

The purpose of the audit is to analyse the on-chain smart contract source code, and to provide basic overview of the project.

While we have used all the information available to us for this straightforward investigation, you should not rely on this report only — we recommend proceeding with several independent audits Be aware that smart contracts deployed on a blockchain aren't secured enough against external vulnerability, or a hack. Be aware that active smart contract owner privileges constitute an elevated impact to smart contract's safety and security. Therefore, SafuAudit does not guarantee the explicit security of the audited smart contract. The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

AUDIT RESULTS

CRITICAL

No critical severity issues have been found.

MEDIUM

No medium severity issues have been found.

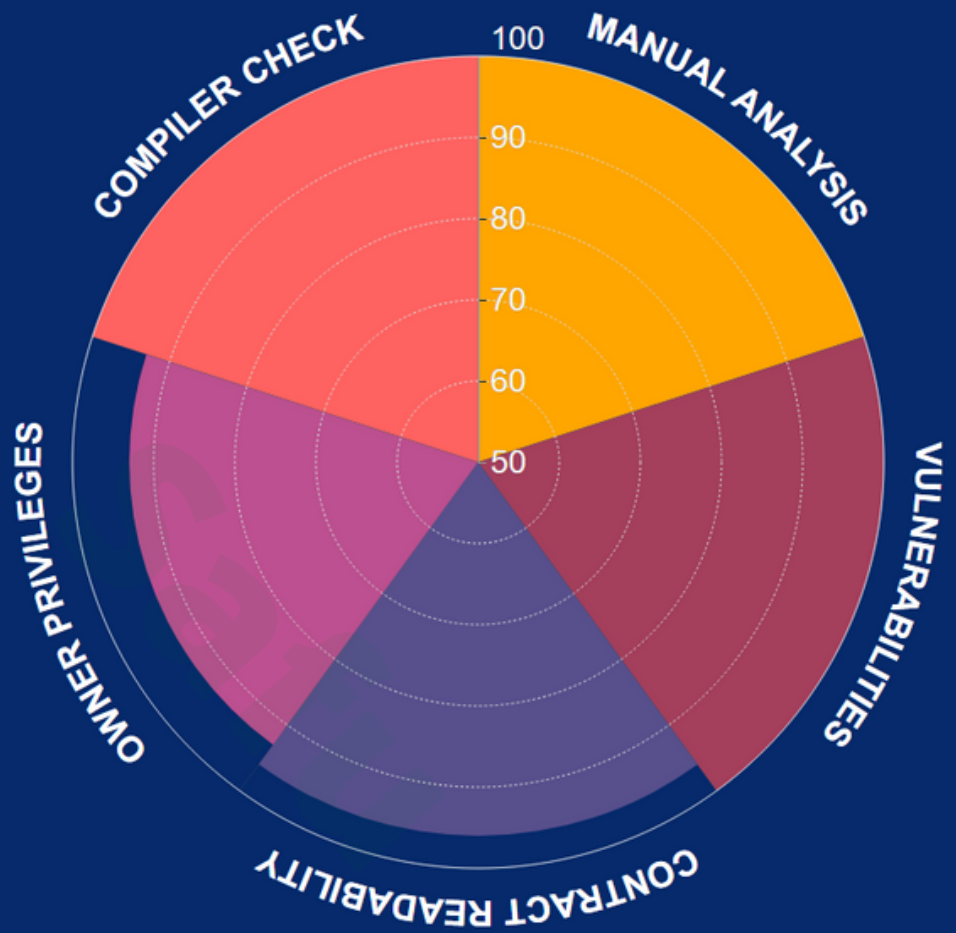
MINOR

No minor severity issues have been found.

INFORMATIONAL

The standard audit model does not offer suggestions and consulting for improvements of efficacity.

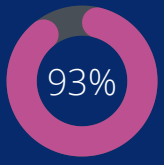
SAFUSCORE



Manual Analysis

Vulnerabilities

Contract Readability



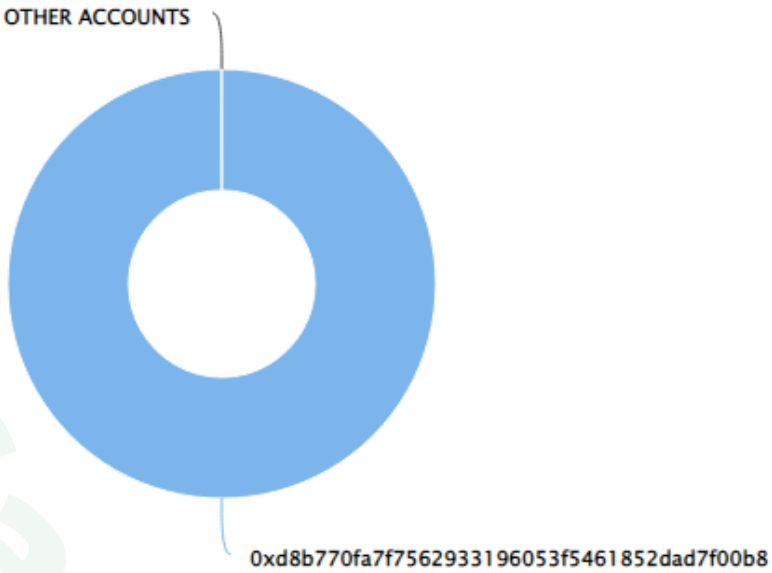
Owner Privileges

Compiler Check

Final Score:97.8

SUMMARY

Top 10 holders



Rank	Address	Quantity (Token)	Percentage
1	0xd8b770fa7f75629331960 53f5461852dad7f00b8	100,000,000	100.0000%

Conclusion

Project GoldMiner does not contain any severe issues or risk characteristics, the owner can modify only the sell fee - up to 15% only.

SafuAudit has tested the security based on manual and automated tests.

Please note that we don't offer any warranties for business model.

SafuAudit.com