Smart Contract Vulnerability Audit

IOTEN

Dec 07, 2021

Smart Contract – Audit Overview

Project Summary

Project Name	IOTEN
Platform	Binance Smart Chain
Language	Solidity
Commits	0xbaae1192b9a115391ed2062b3a5f5d3509461ccb

Audit Summary

Delivery Date	December XX, 2021
Method of Audit	Human and AI
Consultants Engaged	Two
Timeline	December 05, 2021 – December 08, 2021

Vulnerability Summary

Vulnerability Level	Total	Resolved
Critical	0	✓
Major	0	✓
Medium	0	✓
Minor	0	✓

Smart Contract - Contract Overview

All information is recorded as of 12/07/2021.

Contract Name	ERC20.sol
Contract Ticker	IOTN
Contract Address	0xBAAe1192b9A115391Ed2062B3a5f5d3509461CCb
Contract Creator	0x660f8CB3eEbF1eb8d5CCb39C632E335f1E91282F
Decimals	18
Total Supply	10,000,000,000
Token Holders	1
Token Transfers	1
Complier Version	v0.8.9+commit.e5eed63a
Source Code	Solidity
Optimization Enabled	No with 200 runs
Other Settings	default evmVersion, Unlicense license

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Function Default Visibility	✓	✓		✓
Integer Overflow and Underflow	✓	✓		✓
Outdated Compiler Version	✓	✓		✓
Floating Pragma	✓	✓		✓
Unchecked Call Return Value	✓	✓		✓
Unprotected Ether Withdrawal	✓	✓		✓
Unprotected SELFDESTRUCT Instruction	✓	✓		✓
Unencrypted Private Data On-Chain	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Reentrancy	✓	✓		✓
State Variable Default Visibility	✓	✓		✓
Uninitialized Storage Pointer	✓	✓		✓
Assert Violation	✓	✓		✓
Use of Deprecated Solidity Functions	✓	✓		✓
Delegatecall to Untrusted Callee	✓	✓		✓
DoS with Failed Call	✓	✓		✓
Code With No Effects	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Transaction Order Dependence	✓	✓		✓
Authorization through tx.origin	✓	✓		✓
Block values as a proxy for time	✓	✓		✓
Signature Malleability	✓	✓		✓
Incorrect Constructor Name	✓	✓		✓
Shadowing State Variables	✓	✓		✓
Weak Sources of Randomness from Chain Attributes	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Missing Protection against Signature Replay Attacks	✓	✓		✓
Lack of Proper Signature Verification	✓	✓		✓
Requirement Violation	✓	✓		✓
Write to Arbitrary Storage Location	✓	✓		✓
Incorrect Inheritance Order	✓	✓		✓
Insufficient Gas Griefing	✓	✓		✓
Arbitrary Jump with Function Type Variable	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
DoS With Block Gas Limit	✓	✓		✓
Typographical Error	✓	✓		✓
Right-To-Left-Override control character	✓	✓		✓
Presence of unused variables	✓	✓		✓
Unexpected Ether balance	✓	✓		✓
Hash Collisions With Multiple Variable Length Arguments	✓	✓		✓
Message call with hardcoded gas amount	✓	✓		✓

Smart Contract - Code Analysis

We did not identify any minor nor significant vulnerabilities within the contract code.

Smart Contract - Contract Functions

+ [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #

+ [Lib] SafeMath
  - [Int] add
  - [Int] sub
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] div
  - [Int] mod
  - [Int] mod

+ Context
  - [Int] _msgSender
  - [Int] _msgData

+ [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Prv] _functionCallWithValue #

+ Ownable (Context)
  - [Pub] <Constructor> #
  - [Pub] owner
  - [Pub] transferOwner #
    - modifiers: onlyOwner

Smart Contract - Contract Functions

+ ERC20 (Context, IERC20, Ownable)
  - [Pub] <Constructor> #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] transfer #
  - [Pub] allowance
  - [Pub] approve #
  - [Pub] transferFrom #
  - [Pub] tokenFromReflection
  - [Ext] setTaxFeePercent #
    - modifiers: onlyOwner
  - [Ext] setBurnFeePercent #
    - modifiers: onlyOwner
  - [Prv] _reflectFee #
  - [Prv] _getValues
  - [Prv] _getTValues
  - [Prv] _getRValues
  - [Prv] _getRate
  - [Prv] _getCurrentSupply
  - [Prv] calculateTaxFee
  - [Prv] calculateBurnFee
  - [Prv] _approve #
  - [Prv] _transfer #
  - [Prv] _transferStandard #

Smart Contract - Owner Functions

- Owner can modify tax fee
- Owner can modify burn fee
- Owner can transfer contract ownership

Smart Contract - Owner Functions

```
function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
    _taxFee = taxFee;
}
```

```
function setBurnFeePercent(uint256 burnFee) external onlyOwner() {
    _burnFee = burnFee;
}
```

```
function transferOwner(address newOwner) public virtual onlyOwner {
    require(newOwner != address(0), "Ownable: new owner is the zero address");
    emit OwnershipTransferred(_owner, newOwner);
    _owner = newOwner;
}
```

Smart Contract –
Mint function

This contract does not contain a mint function. We were unable to locate a mint function within the code.

Smart Contract –
Contract Ownership

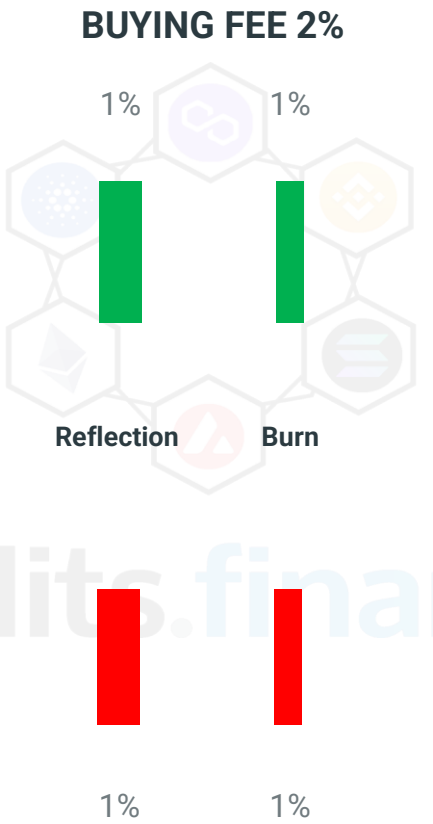
Contract ownership has not been renounced at the time of the audit.
The owner's address is shown as:
0x660f8cb3eebf1eb8d5ccb39c632e335f1e91282f

Smart Contract –
Locked Liquidity

Locked liquidity information was not identified on the website at the time of the audit completion.
Locked liquidity is always be subjected to change.

Smart Contract - Tokenomics

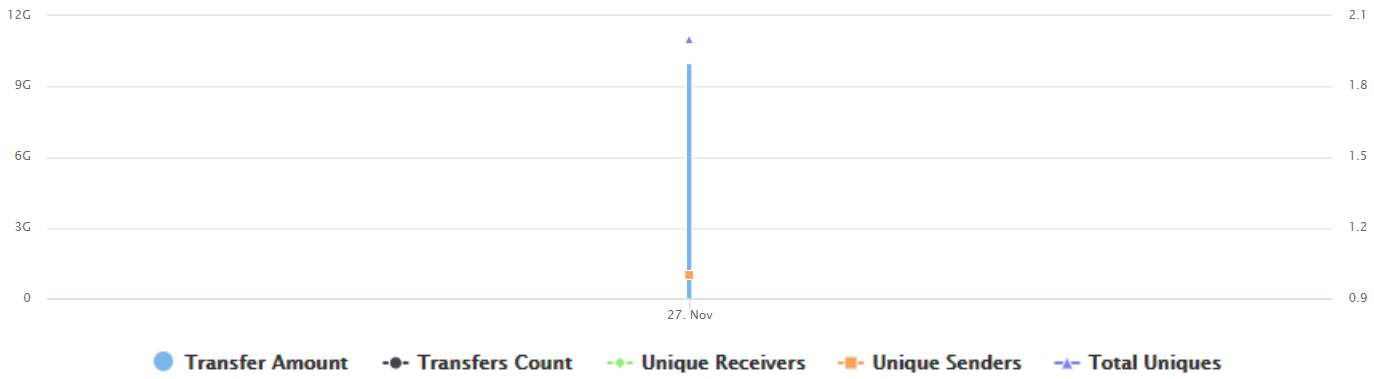
At the time of audit the transaction fees (“tax”) listed below are the fees associated with trading. These fees are taken from every buy and sell transaction unless otherwise stated. Token taxes vary by each project.



Token Holders & Contract Analytics

Top 100 Token Holders

Token Contract Analytics



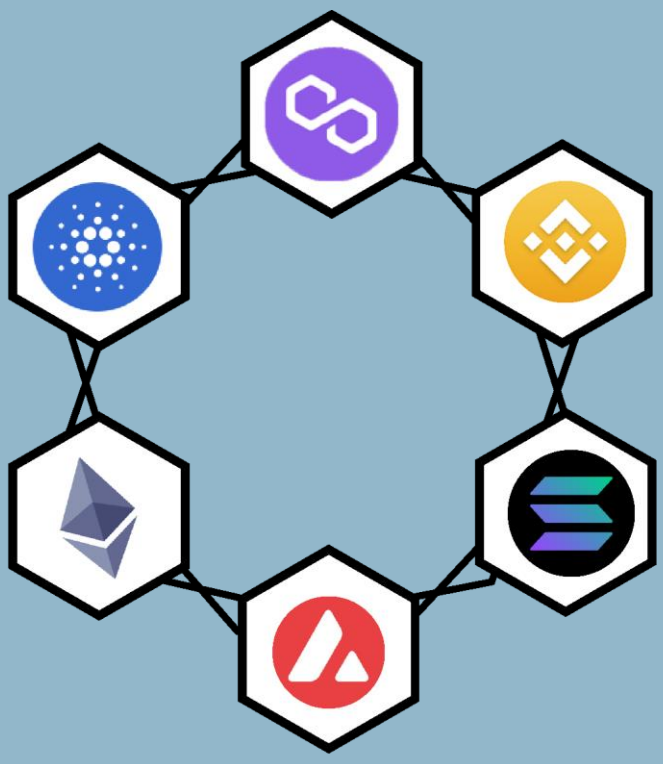
Team Overview

KYC NOT CERTIFIED

Audits.finance has not completed a KYC for the project. Audits.finance has not verified the identity of any team member(s) with government issued ID and photo evidence to match. This project is anonymous.

DISCLAIMER

Audits.finance Inc. is in no way responsible or liable for any legal actions resulting from the use of this presentation. By reading this audit or any part of it, you agree to the terms of this disclaimer. If you do not agree to these terms, please stop reading now, and delete any duplicates of this report. Audits.finance Inc. is an official auditor utilizing the Solidity auditing industry standard. Audits.finance hereby excludes any liability and responsibility. Neither you nor any other person shall have any claim against Audits.finance for any economic loss or damages. Audits.finance Inc. does not guarantee the authenticity of a project, nor does it guarantee the project will not participate in one or any scamming including but not limited to, removing liquidity, selling off team supply, or exit scams. Audits.finance Inc. does not give investment advice in any way. Audits.finance Inc. supplies this presentation for information purposes only, and strongly suggests that none of this information be used as investment advice. Audits.finance in no way endorses or recommends any projects that it audits. Audits.finance is solely responsible for smart contract and project analysis of the projects that it is contracted to audit.
Audits.finance may be contracted by teams, investors, or any other 3rd party in regard to a contract address or project. Audits.finance provides a full report for informational purposes only.



audits.finance

Contact information:
Website: audits.finance
Telegram: auditsfinancegroup
Twitter: auditsfinance
Email: hello@audits.finance