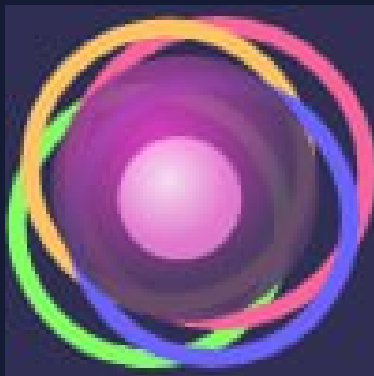


# KISHIELD

Security Audit

**Botrix AI Token**

April 16, 2023



Contract Audited



# Table of Contents

## **1 Audit Summary**

## **2 Project Overview**

### 2.1 Token Summary

### 2.2 Main Contract Assessed

## **3 Smart Contract Vulnerability Checks**

## **4 Contract Ownership**

### 4.1 Privileged Functions

## **5 Important Notes To The Users**

## **6 Findings Summary**

### 6.1 Classification of Issues

### 6.1 Findings Table

01 sendFees will not work as expected during useLaunchFee

02 Unused Variables

03 Variables could be declared as constant

04 Public function that could be declared external

05 Missing events arithmetic

06 Division before Multiplication

## **7 Statistics**

### 7.1 Liquidity

### 7.2 Token Holders

### 7.3 Liquidity Holders

## **8 Liquidity Ownership**

## **9 Disclaimer**



# Audit Summary

This report has been prepared for Botrix AI Token on the BSC network. KISHIELD provides both client-centered and user-centered examination of the smart contracts and their current status when applicable. This report represents the security assessment made to find issues and vulnerabilities on the source code along with the current liquidity and token holder statistics of the protocol.

A comprehensive examination has been performed, utilizing Cross Referencing, Static Analysis, In-House Security Tools, and line-by-line Manual Review.

The auditing process pays special attention to the following considerations:

- Ensuring contract logic meets the specifications and intentions of the client without exposing the user's funds to risk.
- Testing the smart contracts against both common and uncommon attack vectors.
- Inspecting liquidity and holders statistics to inform the current status to both users and client when applicable.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Verifying contract functions that allow trusted and/or untrusted actors to mint, lock, pause, and transfer assets.
- Thorough line-by-line manual review of the entire codebase by industry experts.

# Project Overview

## Token Summary

Parameter	Result
Address	0x623951826f6E5D397CC484b829974d3c8db597d0
Name	Botrix AI
Token Tracker	Botrix AI (BTIX)
Decimals	18
Supply	1,000,000,000
Platform	BSC
compiler	v0.8.17+commit.8df45f5f
Optimization	Yes with 200 runs
LicenseType	None
Language	Solidity
Codebase	<a href="https://bscscan.com/address/0x623951826f6E5D397CC484b829974d3c8db597d0#code">https://bscscan.com/address/0x623951826f6E5D397CC484b829974d3c8db597d0#code</a>
Url	botrix.io

## Main Contract Assessed

Name	Contract	Live
Botrix AI	0x623951826f6E5D397CC484b829974d3c8db597d0	Yes

# Smart Contract Vulnerability Checks

Vulnerability	Automatic Scan	Manual Scan	Result
Unencrypted Private Data On-Chain	Complete	Complete	✓ Low / No Risk
Code With No Effects	Complete	Complete	✓ Low / No Risk
Message call with hardcoded gas amount	Complete	Complete	✓ Low / No Risk
Hash Collisions With Multiple Variable Length Arguments	Complete	Complete	✓ Low / No Risk
Unexpected Ether balance	Complete	Complete	✓ Low / No Risk
Presence of unused variables	Complete	Complete	● Low
Right-To-Left-Override control character (U+202E)	Complete	Complete	✓ Low / No Risk
Typographical Error	Complete	Complete	✓ Low / No Risk
DoS With Block Gas Limit	Complete	Complete	✓ Low / No Risk
Arbitrary Jump with Function Type Variable	Complete	Complete	✓ Low / No Risk
Insufficient Gas Griefing	Complete	Complete	✓ Low / No Risk
Incorrect Inheritance Order	Complete	Complete	✓ Low / No Risk
Write to Arbitrary Storage Location	Complete	Complete	✓ Low / No Risk
Requirement Violation	Complete	Complete	✓ Low / No Risk
Missing Protection against Signature Replay Attacks	Complete	Complete	✓ Low / No Risk
Weak Sources of Randomness from Chain Attributes	Complete	Complete	✓ Low / No Risk

Vulnerability	Automatic Scan	Manual Scan	Result
Authorization through tx.origin	Complete	Complete	✓ Low / No Risk
Delegatecall to Untrusted Callee	Complete	Complete	✓ Low / No Risk
Use of Deprecated Solidity Functions	Complete	Complete	✓ Low / No Risk
Assert Violation	Complete	Complete	✓ Low / No Risk
Reentrancy	Complete	Complete	✓ Low / No Risk
Unprotected SELFDESTRUCT Instruction	Complete	Complete	✓ Low / No Risk
Unprotected Ether Withdrawal	Complete	Complete	✓ Low / No Risk
Unchecked Call Return Value	Complete	Complete	✓ Low / No Risk
Outdated Compiler Version	Complete	Complete	✓ Low / No Risk
Integer Overflow and Underflow	Complete	Complete	✓ Low / No Risk
Function Default Visibility	Complete	Complete	✓ Low / No Risk

## Contract Ownership

The contract ownership of Botrix AI is not currently renounced. The ownership of the contract grants special powers to the protocol creators, making them the sole addresses that can call sensible ownable functions that may alter the state of the protocol.

The current owner is the address 0xC59632bE9F17145e9C810BBD35DbE8d361f8A328 which can be viewed from: [HERE](#)

The owner wallet has the power to call the functions displayed on the privileged functions chart below, if the owner wallet is compromised this privileges could be exploited.

We recommend the team to renounce ownership at the right timing if possible, or gradually migrate to a timelock with governing functionalities in respect of transparency and safety considerations.

## Important Notes To The Users:

- The owner cannot mint tokens after initial deployment.
- The owner cannot stop trading after it is enabled.
- The owner cannot blacklist addresses.
- Address that are exempt free can trade even when the trading is not enabled globally.
- When the owner enables trading for the next 3 blocks a tax of 99% will be used for non tax exempt users.
- Once the owner renounces ownership of the contract, none of the following are applicable.
- The owner can set the sell taxes up to 7%.
- The owner can set the buy taxes up to 7%.
- The 99% period after launch can be increased it to a max of 4 blocks.
- The owner can add/remove addresses from fees.
- The owner can change marketing and dev wallets.
- No high-risk Exploits/Vulnerabilities Were Found in token Source Code.

**Read carefully the notes section and make your own decision before interacting with the audited contract.**

## Audit Passed



# Technical Findings Summary

## Classification of Issues

\*All Issues Found are Informational

Severity	Description
● High	Exploits, vulnerabilities or errors that will certainly or probabilistically lead towards loss of funds, control, or impairment of the contract and its functions. Issues under this classification are recommended to be fixed with utmost urgency
● Medium	Bugs or issues with that may be subject to exploit, though their impact is somewhat limited. Issues under this classification are recommended to be fixed as soon as possible.
● Low	Effects are minimal in isolation and do not pose a significant danger to the project or its users. Issues under this classification are recommended to be fixed nonetheless.
● Info	Consistency, syntax or style best practices. Generally pose a negligible level of risk, if any.

## Findings

Severity	Found
● High	0
● Medium	1
● Low	0
● Info	5
Total	6



# Findings

## sendFees will not work as expected during useLaunchFee

ID	Severity	Contract	Details
01	● Medium	Botrix AI	Function: sendFees

### Description

In the case when useLaunchFee is true, the currentTaxes variable is not assigned any value. This might cause issues if the sendFees function is called, as it expects a Taxes struct as the second argument.

### Recommendation

To avoid this issue, you should ensure that currentTaxes is assigned a value in this scenario or providingLiquidity is set to false.

## Unused Variables

ID	Severity	Contract	Details
02	● Informational	Botrix AI	Mapping _lastSell

### Description

Variables not used as the constructor uses the IERC20Metadata().

### Recommendation

Delete this variables or make use of them in the IERC20Metadata() instead of passing the values directly.

## Variables could be declared as constant

ID	Severity	Contract	Details
03	● Informational	Botrix AI	Variables launchtax

### Description

Gas Optimization. Variables that are never changed could be declared as constant.

### Recommendation

We recommend declaring those variables as constant.

## Public function that could be declared external

ID	Severity	Contract	Details
04	● Informational	Botrix AI	Functions: renounceOwnership, transferOwnership

### Description

Gas Optimization. Public function that could be declared external

### Recommendation

Public functions that are never called by the contract should be declared external to save gas.

## Missing events arithmetic

ID	Severity	Contract	Details
05	● Informational	Botrix AI	Missing events for updateLiquidityTreshhold, updateDeadline

### Description

Functions that change critical arithmetic parameters should emit an event.

### Recommendation

Emit corresponding events for critical parameter changes.

## Division before Multiplication

ID	Severity	Contract	Details
06	● Informational	Botrix AI	Functions: sendFees

### Description

Precision Loss. ``ethToAddLiquidityWith = unitBalance * swapTaxes.liquidity``. Division before multiplication can result in truncation and less accurate results

### Recommendation

Multiplication should be performed before division to not lose precision.

## Privileged Functions (onlyOwner & Others)

Function Name	Parameters	Visibility
renounceOwnership	none	public
transferOwnership	address newOwner	public
sendFees	none	private
updateLiquidityProvide	bool state	external
updateLiquidityTreshhold	uint256 newAmount	external
setBuyTaxes	uint256 _marketing, uint256 _liquidity, uint256 _development	external
setSellTaxes	uint256 _marketing, uint256 _liquidity, uint256 _development	external
enableTrading	none	external
updateDeadline	uint256 _deadline	external
updateMarketingWallet	address newWallet	external
updateDevelopmentWallet	address newWallet	external
updateExemptFee	address _address, bool state	external
bulkExemptFee	calldata accounts, bool state	external
rescueBNB	uint256 weiAmount	external
rescueBEP20	address tokenAdd, uint256 amount	external

# Statistics

## Liquidity Info

Parameter	Result
Pair Address	0x5c868fF843B448D83bf35C6DeAA99DA6b6B9A087
BTIX Reserves	0.00 BTIX
BNB Reserves	0.00 BNB
Liquidity Value	\$0 USD

## Token (BTIX) Holders Info

Parameter	Result
BTIX Percentage Burnt	0.00%
BTIX Amount Burnt	0 BTIX
Top 10 Percentage Own	100.00%
Top 10 Amount Owned	1,000,000,000 BTIX
Top 10 Aprox Value	\$NaN USD

## LP (BTIX/BNB) Holders Info

Parameter	Result
BTIX/BNB % Burnt	0.00%
BTIX/BNB Amount Burnt	0 BTIX/BNB
Top 10 Percentage Owned	0.00%
Top 10 Amount Owned	0 BTIX/BNB
Locked Tokens Percentage	0.00%
Locked Tokens Amount	0 BTIX/BNB

\* All the data displayed above was taken on-chain at block 27397515

\* The tokens on industry-standard burn wallets are not included on the top 10 wallets calculations

# KISHIELD



## Disclaimer

KISHIELD has conducted an independent audit to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the codes that were provided for the scope of this audit. This audit report does not constitute agreement, acceptance or advocacy for the Project that was audited, and users relying on this audit report should not consider this as having any merit for financial advice in any shape, form or nature. The contracts audited do not account for any economic developments that may be pursued by the Project in question, and that the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are completely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment advice, nor should it be used to signal that any persons reading this report should invest their funds without sufficient individual due diligence regardless of the findings presented in this report. Information is provided 'as is', and KISHIELD is under no covenant to the completeness, accuracy or solidity of the contracts audited. In no event will KISHIELD or its partners, employees, agents or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions and/or actions with regards to the information provided in this audit report.

The assessment services provided by KISHIELD is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.