auditace.tech

AuditAce
FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

Kujaku Inu

DATED : 27 MAR 23'

AUDIT SUMMARY

Project name – BNBsongoku

Date: 27 March, 2023

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

Audit Status: Passed
(Contract is Pinksale generated and it follows safu criteria)

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	0	0	0	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 Testnet network:
all tests were done on Bsc Testnet network, each test has its transaction has attached to it.

3- Slither : Static Analysis

Testnet Link:
Contract is a pinksale generated contract (AntiBotLiquidityGeneratorToken) this means all the functionalities has been tested before.

Token Information

Token Name : Kujaku Inu

Token Symbol: KINU

Decimals: 9

Token Supply: 10000000000000

Token Address:
0x3199C857F97C183DDd692Ff0A717E665bE9e1CB3

Checksum:
8fac9d9cc46bb381e5fbd6dab5b11f761e4fdb0e

Owner:
0x49137922C71cbF1C575153B03fC6b5F99a032375
(at the time of audit)

Deployer:
0x49137922C71cbF1C575153B03fC6b5F99a032375

TOKEN OVERVIEW

Fees:

Buy Fees: 3%

Sell Fees: 3%

Transfer Fees: 3%

Fees Privilige: Owner

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Priviliges: updating liquidity threshold - excluding from fees - including in fees - changing fees

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

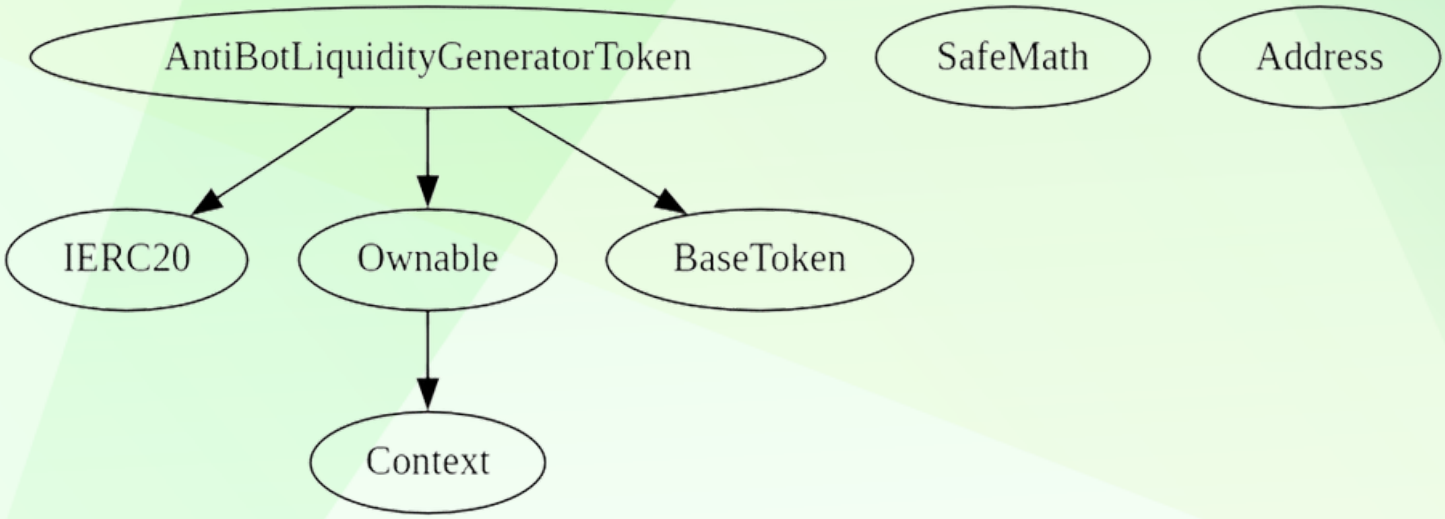
CLASSIFICATION OF RISK

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

INHERITANCE TREE



POINTS TO NOTE

- Owner is not able set able to set more than 25% fees (sum of all fees)
- Owner is not able to disable trades
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to mint new tokens
- Owner is not able to set max wallet/buy/sell amount

CONTRACT ASSESMENT

Contract	Type	Bases		
└	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
Context	Implementation			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
Ownable	Implementation	Context		
└	<Constructor>	Public ❗	🛑	NO❗
└	owner	Public ❗		NO❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
└	_setOwner	Private 🔐	🛑	
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 🔒		
Address	Library			
└	isContract	Internal 🔒		
└	sendValue	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	

CONTRACT ASSESMENT

└	functionCall	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionStaticCall	Internal 🔒		
└	functionStaticCall	Internal 🔒		
└	functionDelegateCall	Internal 🔒	🛑	
└	functionDelegateCall	Internal 🔒	🛑	
└	verifyCallResult	Internal 🔒		
IUniswapV2Router01	Interface			
└	factory	External ❗		NO❗
└	WETH	External ❗		NO❗
└	addLiquidity	External ❗	🛑	NO❗
└	addLiquidityETH	External ❗	💵	NO❗
└	removeLiquidity	External ❗	🛑	NO❗
└	removeLiquidityETH	External ❗	🛑	NO❗
└	removeLiquidityWithPermit	External ❗	🛑	NO❗
└	removeLiquidityETHWithPermit	External ❗	🛑	NO❗
└	swapExactTokensForTokens	External ❗	🛑	NO❗
└	swapTokensForExactTokens	External ❗	🛑	NO❗
└	swapExactETHForTokens	External ❗	💵	NO❗
└	swapTokensForExactETH	External ❗	🛑	NO❗
└	swapExactTokensForETH	External ❗	🛑	NO❗
└	swapETHForExactTokens	External ❗	💵	NO❗
└	quote	External ❗		NO❗
└	getAmountOut	External ❗		NO❗
└	getAmountIn	External ❗		NO❗
└	getAmountsOut	External ❗		NO❗
└	getAmountsIn	External ❗		NO❗
IUniswapV2Router02	Interface	IUniswapV2Router01		
└	removeLiquidityETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗	💵	NO❗
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
IUniswapV2Factory	Interface			
└	feeTo	External ❗		NO❗
└	feeToSetter	External ❗		NO❗
└	getPair	External ❗		NO❗

CONTRACT ASSESMENT

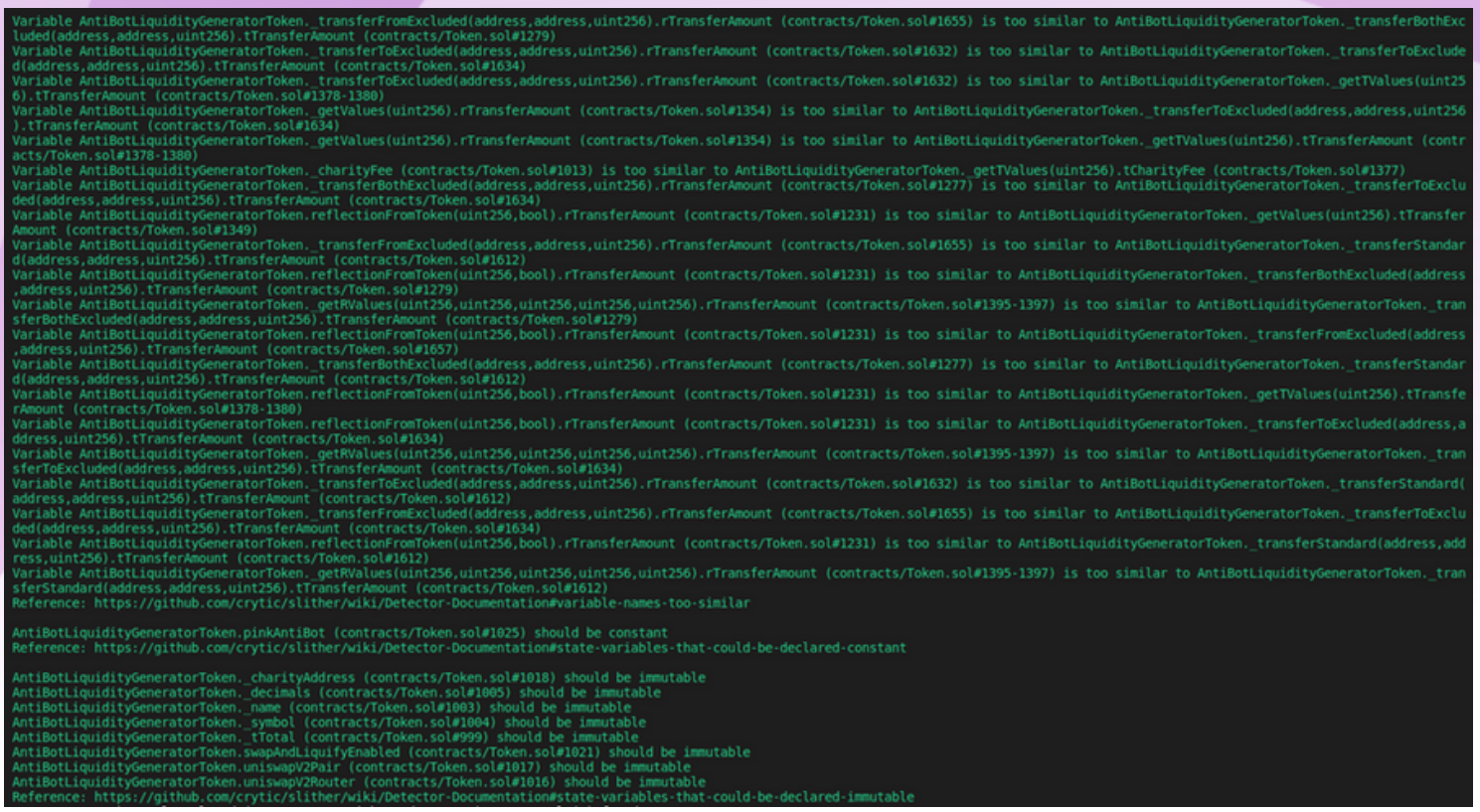
└	allPairs	External ❗			NO❗
└	allPairsLength	External ❗			NO❗
└	createPair	External ❗	🛑		NO❗
└	setFeeTo	External ❗	🛑		NO❗
└	setFeeToSetter	External ❗	🛑		NO❗
IPinkAntiBot	Interface			
└	setTokenOwner	External ❗	🛑		NO❗
└	onPreTransferCheck	External ❗	🛑		NO❗
BaseToken	Implementation			
AntiBotLiquidityGeneratorToken	Implementation	IERC20, Ownable, BaseToken		
└	<Constructor>	Public ❗	💵	NO❗
└	setEnableAntiBot	External ❗	🛑	onlyOwner
└	name	Public ❗			NO❗
└	symbol	Public ❗			NO❗
└	decimals	Public ❗			NO❗
└	totalSupply	Public ❗			NO❗
└	balanceOf	Public ❗			NO❗
└	transfer	Public ❗	🛑		NO❗
└	allowance	Public ❗			NO❗
└	approve	Public ❗	🛑		NO❗
└	transferFrom	Public ❗	🛑		NO❗
└	increaseAllowance	Public ❗	🛑		NO❗
└	decreaseAllowance	Public ❗	🛑		NO❗
└	isExcludedFromReward	Public ❗			NO❗
└	totalFees	Public ❗			NO❗
└	deliver	Public ❗	🛑		NO❗
└	reflectionFromToken	Public ❗			NO❗
└	tokenFromReflection	Public ❗			NO❗
└	excludeFromReward	Public ❗	🛑	onlyOwner
└	includeInReward	External ❗	🛑	onlyOwner
└	_transferBothExcluded	Private 🔐	🛑	
└	excludeFromFee	Public ❗	🛑	onlyOwner
└	setTaxFeePercent	External ❗	🛑	onlyOwner
└	setLiquidityFeePercent	External ❗	🛑	onlyOwner
└	setCharityFeePercent	External ❗	🛑	onlyOwner
└	setSwapBackSettings	External ❗	🛑	onlyOwner
└	<Receive Ether>	External ❗	💵	NO❗
└	_reflectFee	Private 🔐	🛑	

CONTRACT ASSESMENT

└	_getValues	Private 🔐		
└	_getTValues	Private 🔐		
└	_getRValues	Private 🔐		
└	_getRate	Private 🔐		
└	_getCurrentSupply	Private 🔐		
└	_takeLiquidity	Private 🔐	🛑	
└	_takeCharityFee	Private 🔐	🛑	
└	calculateTaxFee	Private 🔐		
└	calculateLiquidityFee	Private 🔐		
└	calculateCharityFee	Private 🔐		
└	removeAllFee	Private 🔐	🛑	
└	restoreAllFee	Private 🔐	🛑	
└	isExcludedFromFee	Public ❗		NO❗
└	_approve	Private 🔐	🛑	
└	_transfer	Private 🔐	🛑	
└	swapAndLiquify	Private 🔐	🛑	lockTheSwap
└	swapTokensForEth	Private 🔐	🛑	
└	addLiquidity	Private 🔐	🛑	
└	_tokenTransfer	Private 🔐	🛑	
└	_transferStandard	Private 🔐	🛑	
└	_transferToExcluded	Private 🔐	🛑	
└	_transferFromExcluded	Private 🔐	🛑	

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

STATIC ANALYSIS



```
Variable AntiBotLiquidityGeneratorToken._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1655) is too similar to AntiBotLiquidityGeneratorToken._transferBothExc
luded(address,address,uint256).tTransferAmount (contracts/Token.sol#1279)
Variable AntiBotLiquidityGeneratorToken._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1632) is too similar to AntiBotLiquidityGeneratorToken._transferToExclude
d(address,address,uint256).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1632) is too similar to AntiBotLiquidityGeneratorToken._getTValues(uint25
6).tTransferAmount (contracts/Token.sol#1378-1380)
Variable AntiBotLiquidityGeneratorToken._getValues(uint256).rTransferAmount (contracts/Token.sol#1354) is too similar to AntiBotLiquidityGeneratorToken._transferToExcluded(address,address,uint256
).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken._getValues(uint256).rTransferAmount (contracts/Token.sol#1354) is too similar to AntiBotLiquidityGeneratorToken._getTValues(uint256).tTransferAmount (contr
acts/Token.sol#1378-1380)
Variable AntiBotLiquidityGeneratorToken._charityFee (contracts/Token.sol#1013) is too similar to AntiBotLiquidityGeneratorToken._getTValues(uint256).tCharityFee (contracts/Token.sol#1377)
Variable AntiBotLiquidityGeneratorToken._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1277) is too similar to AntiBotLiquidityGeneratorToken._transferToExclu
ded(address,address,uint256).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._getValues(uint256).tTransfer
Amount (contracts/Token.sol#1349)
Variable AntiBotLiquidityGeneratorToken._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1655) is too similar to AntiBotLiquidityGeneratorToken._transferStandar
d(address,address,uint256).tTransferAmount (contracts/Token.sol#1612)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._transferBothExcluded(address
,address,uint256).tTransferAmount (contracts/Token.sol#1279)
Variable AntiBotLiquidityGeneratorToken._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#1395-1397) is too similar to AntiBotLiquidityGeneratorToken._tran
sferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1279)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._transferFromExcluded(address
,address,uint256).tTransferAmount (contracts/Token.sol#1657)
Variable AntiBotLiquidityGeneratorToken._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1277) is too similar to AntiBotLiquidityGeneratorToken._transferStandar
d(address,address,uint256).tTransferAmount (contracts/Token.sol#1612)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._getTValues(uint256).tTransfe
rAmount (contracts/Token.sol#1378-1380)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._transferToExcluded(address,a
ddress,uint256).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#1395-1397) is too similar to AntiBotLiquidityGeneratorToken._tran
sferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1632) is too similar to AntiBotLiquidityGeneratorToken._transferStandard(
address,address,uint256).tTransferAmount (contracts/Token.sol#1612)
Variable AntiBotLiquidityGeneratorToken._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1655) is too similar to AntiBotLiquidityGeneratorToken._transferToExclu
ded(address,address,uint256).tTransferAmount (contracts/Token.sol#1634)
Variable AntiBotLiquidityGeneratorToken.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#1231) is too similar to AntiBotLiquidityGeneratorToken._transferStandard(address,add
ress,uint256).tTransferAmount (contracts/Token.sol#1612)
Variable AntiBotLiquidityGeneratorToken._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#1395-1397) is too similar to AntiBotLiquidityGeneratorToken._tran
sferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1612)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

AntiBotLiquidityGeneratorToken.pinkAntiBot (contracts/Token.sol#1025) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

AntiBotLiquidityGeneratorToken._charityAddress (contracts/Token.sol#1018) should be immutable
AntiBotLiquidityGeneratorToken._decimals (contracts/Token.sol#1005) should be immutable
AntiBotLiquidityGeneratorToken._name (contracts/Token.sol#1003) should be immutable
AntiBotLiquidityGeneratorToken._symbol (contracts/Token.sol#1004) should be immutable
AntiBotLiquidityGeneratorToken._tTotal (contracts/Token.sol#999) should be immutable
AntiBotLiquidityGeneratorToken.swapAndLiquifyEnabled (contracts/Token.sol#1021) should be immutable
AntiBotLiquidityGeneratorToken.uniswapV2Pair (contracts/Token.sol#1017) should be immutable
AntiBotLiquidityGeneratorToken.uniswapV2Router (contracts/Token.sol#1016) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
```

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

No issues found

FUNCTIONAL TESTING

Contract is a pinksale generated contract (**AntiBotLiquidityGeneratorToken**) this means all the functionalities has been tested before.

TOKEN DISTRIBUTION

it should be noted that at time of writing this audit, there are 2 holders of **KINU** token:

0x49137922c71cbf1c575153b03fc6b5f99a032375 => holds 50% of supply

0x94d930f04cca4145f8a198eea67e0cde29a2e4ec => holds 50% of supply

MANUAL TESTING

No Issues Found

Social Media Overview

Here are the Social Media Accounts of Kujaku Inu

https://t.me/Kujakulnu

https://twitter.com/Kujakulnu

www.kujakuinu.club

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