auditace.tech

AuditAce
FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

Ape Inu

DATED : 7 June 23'

AUDIT SUMMARY

Project name – Ape Inu

Date: 7 June, 2023

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

Audit Status: **Passed**

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 Test Network:
all tests were done on BSC Test network, each test has its transaction has attached to it.

3- Slither : Static Analysis

Testnet Link: all tests were done using this contract, tests are done on BSC Testnet

https://testnet.bscscan.com/token/0x8910af5a07dA02d0FC44cc10a0A10794ED7682d1

Token Information

Token Name : Ape Inu

Token Symbol: APEINU

Decimals: 9

Token Supply:1,000,000,000

Token Address:
0xF85C425bEeFbDA8d718DFbbE70a363F2e4978E10

Checksum:
58224918f35dc30c31d40d08241278a580432124

Owner: -
0xE63937C960dD8aA91369F910afc73ce811bea665

Deployer:
0xE63937C960dD8aA91369F910afc73ce811bea665

TOKEN OVERVIEW

Fees:

Buy Fees: 0-8%

Sell Fees: 0-8%

Transfer Fees: 0-8%

Fees Privilige: Owner

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: none

Blacklist: No

Other Priviliges: changing fees

- excluding from fees
- including in fees
- initial distribution of the tokens

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

- [x] Return values of low-level calls
- [x] Private modifier
- [x] Multiple Sends
- [x] Using Suicide
- [x] Gas Limitand Loops
- [x] Address hardcoded
- [x] Exception Disorder
- [x] Using inline assembly
- [x] Divide before multiply
- [x] Missing Zero Address Validation
- [x] Compiler version not fixed
- [x] **Gasless Send**
- [x] Using block.timestamp
- [x] Re-entrancy
- [x] Tautology or contradiction
- [x] Timestamp Dependence
- [x] Revert/require functions
- [x] Use of tx.origin
- [x] Integer overflow/underflow
- [x] Dangerous strict equalities
- [x] Using SHA3
- [x] Using throw

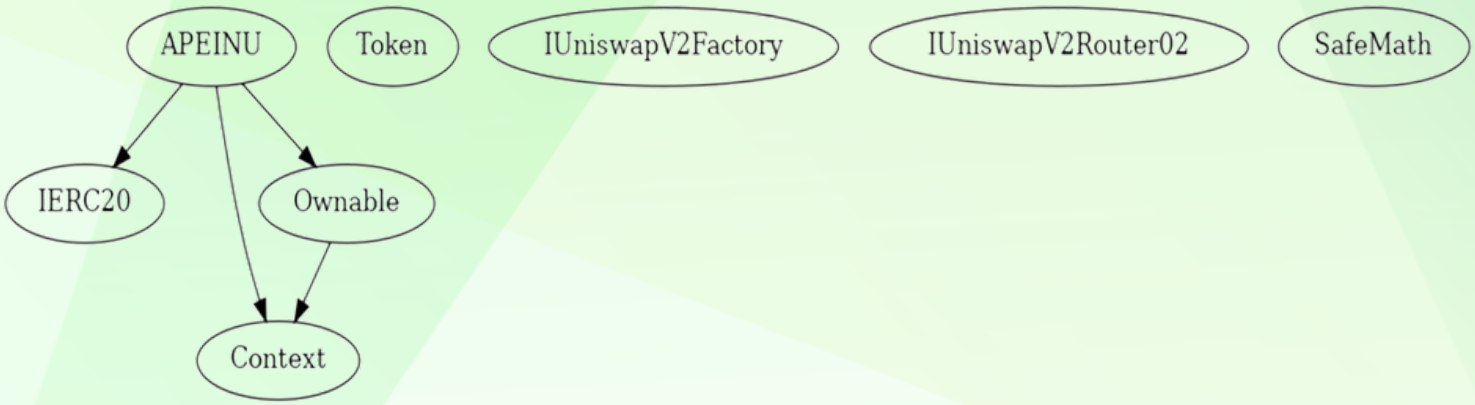
CLASSIFICATION OF RISK

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 able to set buy/sell fees up to 8% each (16% total)
- Owner is not ablet o set transfer taxes (0% forever)
- Owner is not able to set max buy/sell/transfer/hold amount
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to disable trades
- Owner is not able to mint new tokens

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 ❗
Token	Interface			
└	transferFrom	External ❗	🛑	NO ❗
└	transfer	External ❗	🛑	NO ❗
IUniswapV2Factory	Interface			
└	createPair	External ❗	🛑	NO ❗
IUniswapV2Router02	Interface			
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗	🛑	NO ❗
└	factory	External ❗		NO ❗
└	WETH	External ❗		NO ❗
└	addLiquidityETH	External ❗	💵	NO ❗
Context	Implementation			
└	_msgSender	Internal 🔒		
SafeMath	Library			
└	add	Internal 🔒		
└	sub	Internal 🔒		
└	sub	Internal 🔒		
└	mul	Internal 🔒		
└	div	Internal 🔒		
└	div	Internal 🔒		
Ownable	Implementation	Context		
└	<Constructor>	Public ❗	🛑	NO ❗
└	owner	Public ❗		NO ❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
APEINU	Implementation	Context, IERC20, Ownable		
└	<Constructor>	Public ❗	🛑	NO ❗

CONTRACT ASSESMENT

└	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 ❗
└	tokenFromReflection	Private 🔐		
└	_approve	Private 🔐	🛑	
└	_transfer	Private 🔐	🛑	
└	swapTokensForEth	Private 🔐	🛑	lockTheSwap
└	sendETHToFee	Private 🔐	🛑	
└	_tokenTransfer	Private 🔐	🛑	
└	rescueForeignTokens	Public ❗	🛑	onlyDev
└	setNewDevAddress	Public ❗	🛑	onlyDev
└	setNewMarketingAddress	Public ❗	🛑	onlyDev
└	_transferStandard	Private 🔐	🛑	
└	_takeTeam	Private 🔐	🛑	
└	_reflectFee	Private 🔐	🛑	
└	<Receive Ether>	External ❗	💵	NO ❗
└	_getValues	Private 🔐		
└	_getTValues	Private 🔐		
└	_getRValues	Private 🔐		
└	_getRate	Private 🔐		
└	_getCurrentSupply	Private 🔐		
└	manualswap	External ❗	🛑	NO ❗
└	manualsend	External ❗	🛑	NO ❗
└	setFee	Public ❗	🛑	onlyDev
└	toggleSwap	Public ❗	🛑	onlyDev
└	excludeMultipleAccountsFromFees	Public ❗	🛑	onlyOwner

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

STATIC ANALYSIS

```
Reentrancy in APEINU.transferFrom(address,address,uint256) (contracts/Token.sol#220-228):
        External calls:
        - _transfer(sender,recipient,amount) (contracts/Token.sol#221)
                - _developmentAddress.transfer(amount.div(2)) (contracts/Token.sol#292)
                - _marketingAddress.transfer(amount.div(2)) (contracts/Token.sol#293)
        State variables written after the call(s):
        - _approve(sender,_msgSender(),_allowances[sender][_msgSender()].sub(amount,ERC20: transfer amount exceeds allowance)) (contracts/Token.sol#222-226)
                - _allowances[owner][spender] = amount (contracts/Token.sol#239)
        Event emitted after the call(s):
        - Approval(owner,spender,amount) (contracts/Token.sol#240)
                - _approve(sender,_msgSender(),_allowances[sender][_msgSender()].sub(amount,ERC20: transfer amount exceeds allowance)) (contracts/Token.sol#222-
226)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-4

Variable APEINU._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#324) is too similar to APEINU._transferStandard(address,address,
uint256).tTransferAmount (contracts/Token.sol#324)
Variable APEINU._getValues(uint256).rTransferAmount (contracts/Token.sol#349) is too similar to APEINU._transferStandard(address,address,uint256).tTransferAmoun
t (contracts/Token.sol#324)
Variable APEINU._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#372) is too similar to APEINU._getValues(uint256).tTransferAmo
unt (contracts/Token.sol#347)
Variable APEINU._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#372) is too similar to APEINU._transferStandard(address,addres
s,uint256).tTransferAmount (contracts/Token.sol#324)
Variable APEINU._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#372) is too similar to APEINU._getTValues(uint256,uint256,uint
256).tTransferAmount (contracts/Token.sol#360)
Variable APEINU._getValues(uint256).rTransferAmount (contracts/Token.sol#349) is too similar to APEINU._getTValues(uint256,uint256,uint256).tTransferAmount (con
tracts/Token.sol#360)
Variable APEINU._getValues(uint256).rTransferAmount (contracts/Token.sol#349) is too similar to APEINU._getValues(uint256).tTransferAmount (contracts/Token.sol#
347)
Variable APEINU._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#324) is too similar to APEINU._getTValues(uint256,uint256,uint25
6).tTransferAmount (contracts/Token.sol#360)
Variable APEINU._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#324) is too similar to APEINU._getValues(uint256).tTransferAmoun
t (contracts/Token.sol#347)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

APEINU._tOwned (contracts/Token.sol#129) is never used in APEINU (contracts/Token.sol#125-423)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variable

APEINU.uniswapV2Pair (contracts/Token.sol#155) should be immutable
APEINU.uniswapV2Router (contracts/Token.sol#154) 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 major issues were found in the output

FUNCTIONAL TESTING

1- Adding liquidity (passed):
https://testnet.bscscan.com/tx/0x7a1aad3e4d7a3645b383de41814d1894c58415e41f3a68bfba6c27cc8710c6b6

2- Buying when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x8ec344ee898c8770ad5d2c869d87de31ef26d65658cbd915aaac46c2c236f54b

3- Selling when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x8ec344ee898c8770ad5d2c869d87de31ef26d65658cbd915aaac46c2c236f54b

4- Transferring when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x03c10a53c1aac138863436eb8a2f8163d795b839098497ab0f043ef1c5ac2d52

5- Buying when not excluded from fees (0-8% tax) (passed):
https://testnet.bscscan.com/tx/0x332b8f300a5da8bb6278e67ebb21cdf3a2391b589aff3dcf4b13fc4d6c82f1b6

6- Selling when not excluded from fees (0-8% tax) (passed):
https://testnet.bscscan.com/tx/0x6b652da2cd1ebc9374f6df06373994d39a7c33817eee839f8f139cfa3ff63661

FUNCTIONAL TESTING

7- Transferring when not excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x2f66a7fa0a6f1718605fe00422231770da7332882b0ba964cfe7e8f92c928102

8- Internal swap (passed):
Marketing and development wallets received BNB
https://testnet.bscscan.com/address/0xa6743c504122b5f128120C4018b3c3e3E2814164#internaltx

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