# AuditAce
## FROM INCEPTION TO SUCCESS

# Smart Contract Audit

**FOR**

# PEPE Chinese Whales

**DATED : 10 May 23'**

# AUDIT SUMMARY

**Project name** – PEPE Chinese Whales

**Date**: 10  May, 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 | 1 |
| 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/0x7c9b63a1d8ce c9223ac651a684199718436bf8c7

# Token Information

**Token Name** : PEPE Chinese Whales

**Token Symbol**: PEPECHI

**Decimals**: 18

**Token Supply**:888,888,888,888

**Token Address:**
0xece0b8Cf584276D30B42ef9D52e8599D55aFAbDA

**Checksum:**
a447fe90e9bbae090a21d11181991884f6717cf2

**Owner:**
0x7eE9E05143D7202d6870490199a9675D97BD2016

# TOKEN OVERVIEW

**Fees:**

Buy Fees:  0 %

Sell Fees:  0 %

Transfer Fees: 0%

**Fees Privilige:** none

**Ownership** :  none

**Minting:** No mint function

**Max Tx Amount/ Max Wallet Amount:** No

**Blacklist: No**

**Other Priviliges**:  none

# 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

◆ **Critical**

◆ **High-Risk**

◆ **Medium-Risk**

◆ **Low-Risk**

◆ **Gas Optimization /Suggestion**

## Description

These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.

A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.

A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.

A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.

A vulnerability that has an informational character but is not affecting any of the code.

# Findings

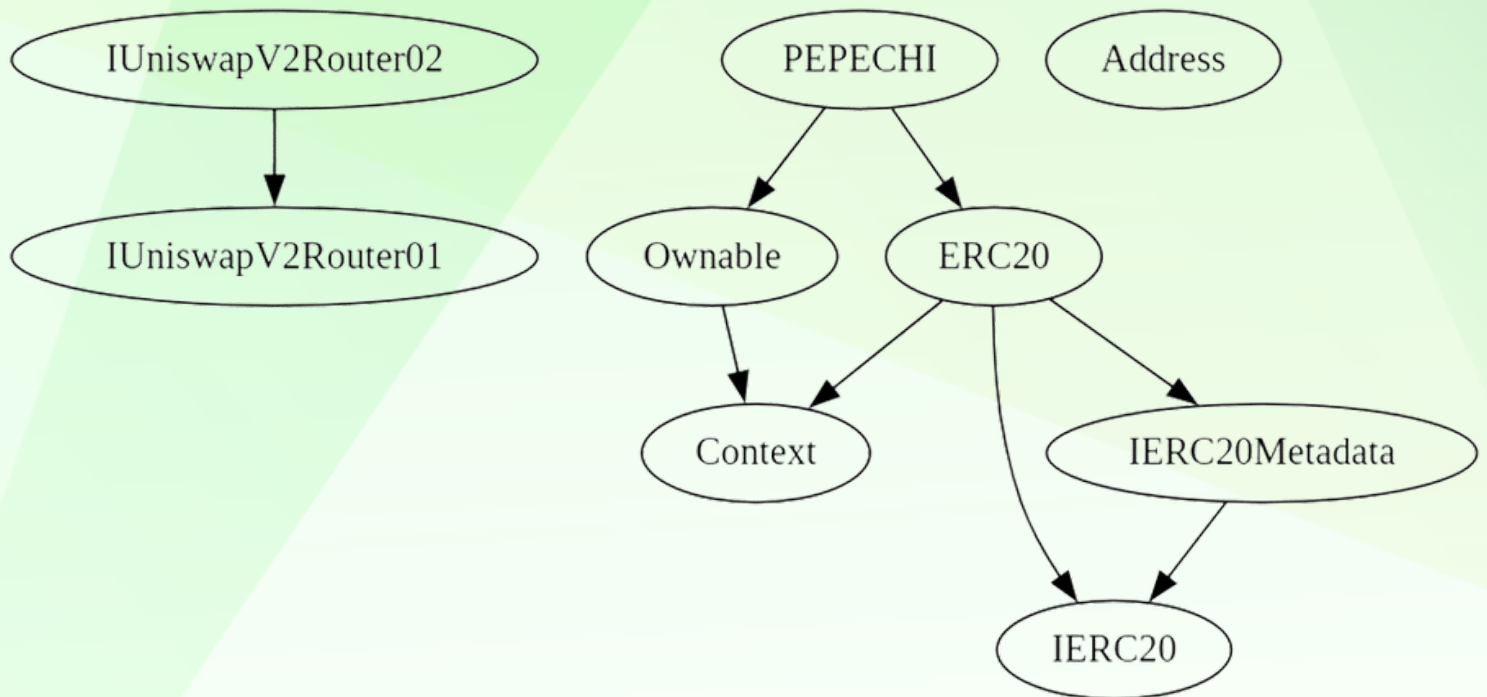| Severity | Found |
|---|---|
| ◆ **Critical** | 0 |
| ◆ **High-Risk** | 0 |
| ◆ **Medium-Risk** | 0 |
| ◆ **Low-Risk** | 0 |
| ◆ **Gas Optimization / Suggestions** | 1 |

# INHERITANCE TREE

# POINTS TO NOTE

- Owner is not able to set set buy/sell/transfer tax (0% always)
- Owner is not able to set a max buy/transfer/wallet/sell 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
- **Owner must enable trades for holders to be able to trade**

# CONTRACT ASSESMENT

| Contract | Type | Bases | | |
|:----------:|:------------------:|:----------------:|:----------------:|:----------------:|
| └ | **Function Name** | **Visibility** | **Mutability** | **Modifiers** |
| **IUniswapV2Factory** | Interface | | | |
| └ | feeTo | External ❗ | | NO❗ |
| └ | feeToSetter | External ❗ | | NO❗ |
| └ | getPair | External ❗ | | NO❗ |
| └ | allPairs | External ❗ | | NO❗ |
| └ | allPairsLength | External ❗ | | NO❗ |
| └ | createPair | External ❗ | 🛑 | NO❗ |
| └ | setFeeTo | External ❗ | 🛑 | NO❗ |
| └ | setFeeToSetter | External ❗ | 🛑 | NO❗ |
| | | | | |
| **IUniswapV2Pair** | Interface | | | |
| └ | name | External ❗ | | NO❗ |
| └ | symbol | External ❗ | | NO❗ |
| └ | decimals | External ❗ | | NO❗ |
| └ | totalSupply | External ❗ | | NO❗ |
| └ | balanceOf | External ❗ | | NO❗ |
| └ | allowance | External ❗ | | NO❗ |
| └ | approve | External ❗ | 🛑 | NO❗ |
| └ | transfer | External ❗ | 🛑 | NO❗ |
| └ | transferFrom | External ❗ | 🛑 | NO❗ |
| └ | DOMAIN_SEPARATOR | External ❗ | | NO❗ |
| └ | PERMIT_TYPEHASH | External ❗ | | NO❗ |
| └ | nonces | External ❗ | | NO❗ |
| └ | permit | External ❗ | 🛑 | NO❗ |
| └ | MINIMUM_LIQUIDITY | External ❗ | | NO❗ |
| └ | factory | External ❗ | | NO❗ |
| └ | token0 | External ❗ | | NO❗ |
| └ | token1 | External ❗ | | NO❗ |
| └ | getReserves | External ❗ | | NO❗ |
| └ | price0CumulativeLast | External ❗ | | NO❗ |
| └ | price1CumulativeLast | External ❗ | | NO❗ |
| └ | kLast | External ❗ | | NO❗ |
| └ | mint | External ❗ | 🛑 | NO❗ |
| └ | burn | External ❗ | 🛑 | NO❗ |
| └ | swap | External ❗ | 🛑 | NO❗ |
| └ | skim | External ❗ | 🛑 | NO❗ |
| └ | sync | External ❗ | 🛑 | NO❗ |
| └ | initialize | External ❗ | 🛑 | NO❗ |

# CONTRACT ASSESMENT

| | | | | |
| **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❗ |
| | | | | |
| **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❗ |
| | | | | |

# CONTRACT ASSESMENT

| **Address** | Library | ||||
| └ | isContract | Internal 🔒 | | |
| └ | sendValue | Internal 🔒 | 🛑 | |
| └ | functionCall | Internal 🔒 | 🛑 | |
| └ | functionCall | Internal 🔒 | 🛑 | |
| └ | functionCallWithValue | Internal 🔒 | 🛑 | |
| └ | functionCallWithValue | Internal 🔒 | 🛑 | |
| └ | functionStaticCall | Internal 🔒 | | |
| └ | functionStaticCall | Internal 🔒 | | |
| └ | functionDelegateCall | Internal 🔒 | 🛑 | |
| └ | functionDelegateCall | Internal 🔒 | 🛑 | |
| └ | verifyCallResultFromTarget | Internal 🔒 | | |
| └ | verifyCallResult | Internal 🔒 | | |
| └ | _revert | Private 🔐 | | |
||||||
| **Context** | Implementation | ||||
| └ | _msgSender | Internal 🔒 | | |
| └ | _msgData | Internal 🔒 | | |
||||||
| **Ownable** | Implementation | Context ||||
| └ | <Constructor> | Public ❗ | 🛑 | NO❗ |
| └ | owner | Public ❗ | | NO❗ |
| └ | renounceOwnership | Public ❗ | 🛑 | onlyOwner |
| └ | transferOwnership | Public ❗ | 🛑 | onlyOwner |
||||||
| **ERC20** | Implementation | Context, IERC20, IERC20Metadata ||||
| └ | <Constructor> | Public ❗ | 🛑 | NO❗ |
| └ | 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❗ |
| └ | _transfer | Internal 🔒 | 🛑 | |
| └ | _mint | Internal 🔒 | 🛑 | |
| └ | _burn | Internal 🔒 | 🛑 | |

# CONTRACT ASSESMENT

| └ | _approve | Internal 🔒 | 🛑 | | |
| └ | _beforeTokenTransfer | Internal 🔒 | 🛑 | | |
| └ | _afterTokenTransfer | Internal 🔒 | 🛑 | | |
| | | | | | |
| **PEPECHI** | Implementation | ERC20, Ownable | | |
| └ | <Constructor> | Public ❗ | 🛑 | ERC20 |
| └ | <Receive Ether> | External ❗ | 💵 | NO❗ |
| └ | claimStuckTokens | External ❗ | 🛑 | onlyOwner |
| └ | excludeFromFees | External ❗ | 🛑 | onlyOwner |
| └ | isExcludedFromFees | Public ❗ | | NO❗ |
| └ | enableTrading | External ❗ | 🛑 | onlyOwner |
| └ | _transfer | Internal 🔒 | 🛑 | |
| └ | setSwapEnabled | External ❗ | 🛑 | onlyOwner |
| └ | setSwapTokensAtAmount | External ❗ | 🛑 | onlyOwner |
| └ | swapAndSendMarketing | Private 🔐 | 🛑 | |

Legend

| Symbol | Meaning |
|:--------:|-----------|
| 🛑 | Function can modify state |
| 💵 | Function is payable |

# STATIC ANALYSIS

```
Address._revert(bytes,string) (contracts/Token.sol#560-575) is never used and should be removed
Address.functionCall(address,bytes) (contracts/Token.sol#419-430) is never used and should be removed
Address.functionCall(address,bytes,string) (contracts/Token.sol#432-438) is never used and should be removed
Address.functionCallWithValue(address,bytes,uint256) (contracts/Token.sol#440-452) is never used and should be removed
Address.functionCallWithValue(address,bytes,uint256,string) (contracts/Token.sol#454-474) is never used and should be removed
Address.functionDelegateCall(address,bytes) (contracts/Token.sol#503-513) is never used and should be removed
Address.functionDelegateCall(address,bytes,string) (contracts/Token.sol#515-528) is never used and should be removed
Address.functionStaticCall(address,bytes) (contracts/Token.sol#476-486) is never used and should be removed
Address.functionStaticCall(address,bytes,string) (contracts/Token.sol#488-501) is never used and should be removed
Address.isContract(address) (contracts/Token.sol#402-404) is never used and should be removed
Address.verifyCallResult(bool,bytes,string) (contracts/Token.sol#548-558) is never used and should be removed
Address.verifyCallResultFromTarget(address,bool,bytes,string) (contracts/Token.sol#530-546) is never used and should be removed
Context._msgData() (contracts/Token.sol#583-586) is never used and should be removed
ERC20._burn(address,uint256) (contracts/Token.sol#773-788) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code

Pragma version^0.8.17 (contracts/Token.sol#34) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6/0.8.16
solc-0.8.19 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity

Low level call in Address.sendValue(address,uint256) (contracts/Token.sol#406-417):
        - (success) = recipient.call{value: amount}() (contracts/Token.sol#415)
Low level call in Address.functionCallWithValue(address,bytes,uint256,string) (contracts/Token.sol#454-474):
        - (success,returndata) = target.call{value: value}(data) (contracts/Token.sol#464-466)
Low level call in Address.functionStaticCall(address,bytes,string) (contracts/Token.sol#488-501):
        - (success,returndata) = target.staticcall(data) (contracts/Token.sol#493)
Low level call in Address.functionDelegateCall(address,bytes,string) (contracts/Token.sol#515-528):
        - (success,returndata) = target.delegatecall(data) (contracts/Token.sol#520)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-calls

Function IUniswapV2Pair.DOMAIN_SEPARATOR() (contracts/Token.sol#96) is not in mixedCase
Function IUniswapV2Pair.PERMIT_TYPEHASH() (contracts/Token.sol#98) is not in mixedCase
Function IUniswapV2Pair.MINIMUM_LIQUIDITY() (contracts/Token.sol#129) is not in mixedCase
Function IUniswapV2Router01.WETH() (contracts/Token.sol#169) is not in mixedCase
Parameter PEPECHI.setSwapEnabled(bool)._enabled (contracts/Token.sol#970) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions

Redundant expression "this (contracts/Token.sol#584)" inContext (contracts/Token.sol#578-587)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundant-statements

Variable IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (contracts/Token.sol#174) is too similar to IUniswapV
2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBDesired (contracts/Token.sol#175)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

PEPECHI.creator (contracts/Token.sol#823) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

PEPECHI.marketingFeeOnBuy (contracts/Token.sol#825) should be immutable
PEPECHI.marketingFeeOnSell (contracts/Token.sol#826) should be immutable
PEPECHI.marketingWallet (contracts/Token.sol#828) should be immutable
PEPECHI.uniswapV2Pair (contracts/Token.sol#819) should be immutable
PEPECHI.uniswapV2Router (contracts/Token.sol#818) 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

**Router (PCS V2):**
0xD99D1c33F9fC3444f8101754aBC46c52416550D1

All the functionalities have been tested, no issues were found

**1- Adding liquidity (passed):**
https://testnet.bscscan.com/tx/0x29122764432241be8006bf43f7c
a4ea62ffe95371feef836c4e049584a0b37df

**2- Buying when excluded (0% tax) (passed):**
https://testnet.bscscan.com/tx/0x422263709341de141085dfe879c
260fd02162b48893dbb8c3fbf57c05eca5c22

**3- Selling when excluded (0% tax) (passed):**
https://testnet.bscscan.com/tx/0x4dda854eed5f22145d5392b3ee
13db4fcd01f496f3f023e30a72ff7bc84a95e5

**4- Transferring when excluded (0% tax) (passed):**
https://testnet.bscscan.com/tx/0x681eef7a23be41b7e4178c8a3dd
d19ff90740a5292467f076f520f6962ec5739

**5- Buying when not excluded (0% tax) (passed):**
https://testnet.bscscan.com/tx/0xfd494072ece997152eed038bcf9
d76ec9c5fa69a10e887394fecd7f78b163bc8

**6- Selling when not excluded (0% tax) (passed):**
https://testnet.bscscan.com/tx/0x1588a3f5872a266e55f7c53710
50b49bb7f929b8ca89840931623d0889c8d085

# FUNCTIONAL TESTING

**7- Transferring when not excluded (0% tax)** <span style="color:green">**(passed):**</span>
https://testnet.bscscan.com/tx/0x95c79b9adcb1e8b127bff75b3e34557f63e49cb377c3d355473f464511cbfa12

# MANUAL TESTING

## Informational – Redundant code

**Status:** Not Resolved

**Overview:**

Marketing buy and sell tax is zero and can not be changed later, this means some features in the contract (like internal swap) are not needed until there are no tokens in the contract.

**Suggestion:**

delete sections of the code that are related to tax (like internal swap, etc...) in order to reduce overall gas usage

# 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**