expelee

Building the Futuristic Blockchain Ecosystem

Security Audit Report

FOR

Pepe Inu

OVERVIEW

The Expelee team has performed a line-by-line manual analysis and automated review of the smart contract. The smart contract was analysed mainly for common smart contract vulnerabilities, exploits, and manipulation hacks. According to the smart contract audit:

Audit Result	Passed
KYC Verification	Not Done
Audit Date	13 march 2023

PROJECT DESCRIPTION

Pepe Inu

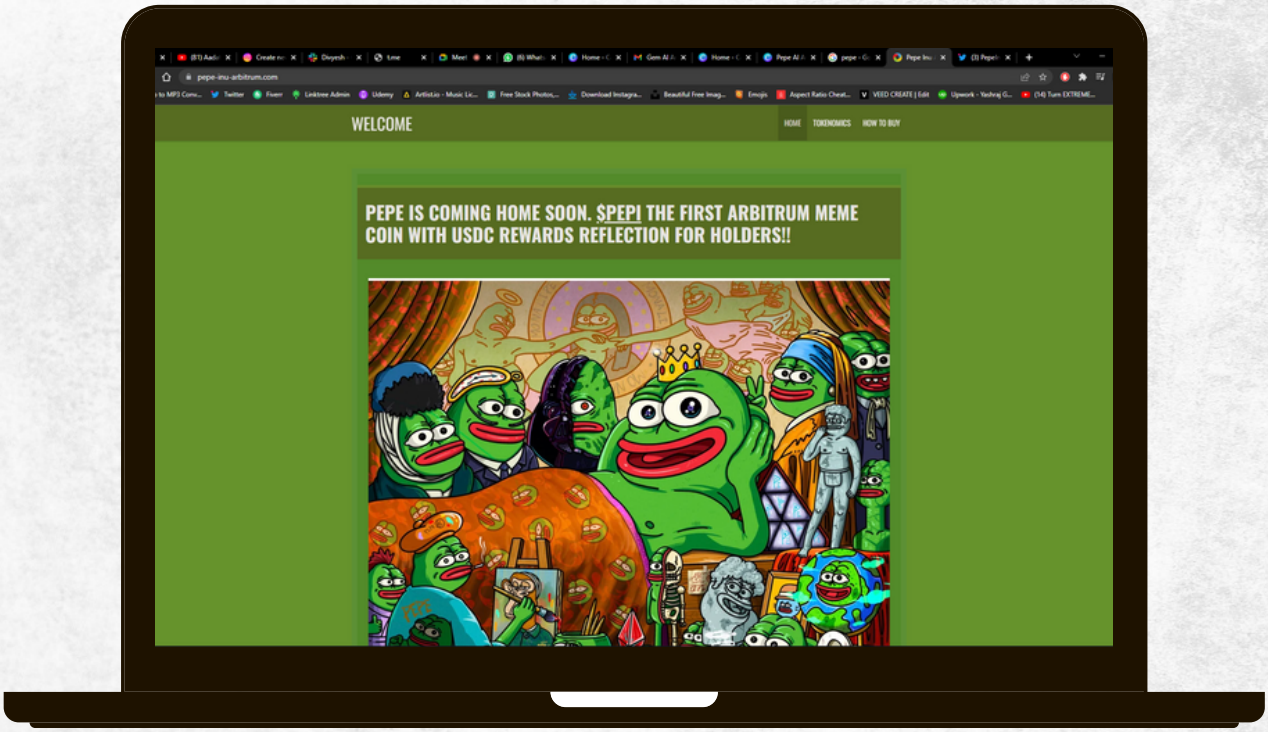
The PEPE INU aims to introduce individuals to the Arbitrum ecosystem through its fun and friendly Pepe! The project is community owned and managed, with strict limits on individual holdings to promote decentralization and widespread adoption. By leveraging the power of memes and community-driven content creation, PEPE INU aims to increase awareness and understanding of the Arbitrum universe, serving as a guide and companion to all Arbinauts along the way!

expelee

expelee.com

Social Media Profiles

Pepe Inu



https://www.pepe-inu-arbitrum.com/

https://t.me/PepeInuArbi

https://twitter.com/Pepe_Inu_Arbi

It's always good to check the social profiles of the project, before making your investment.

-Team Expelee

CONTRACT DETAILS

Token Name

Pepe Inu

Symbol

Pepi

Network

BSC

Language

Solidity

Contract Address (Verified)

0x29585223035a8bF020567CD188834bA727a4B4a9

Token Type

BEP20

Total Supply

1,000,000,000,000

Contract SHA-256 Checksum:

900adda7e7785dba36e8506d52cbc16c27be1460

Owner's Wallet

0x179F1dd0379bA992799e23F5013A570e00f5e198

Deployer's Wallet

0x179F1dd0379bA992799e23F5013A570e00f5e198

AUDIT METHODOLOGY

Audit Details

Our comprehensive audit report provides a full overview of the audited system's architecture, smart contract codebase, and details on any vulnerabilities found within the system.

Audit Goals

The audit goal is to ensure that the project is built to protect investors and users, preventing potentially catastrophic vulnerabilities after launch, that lead to scams and rugpulls.

Code Quality

Our analysis includes both automatic tests and manual code analysis for the following aspects:

- Exploits
- Back-doors
- Vulnerability
- Accuracy
- Readability

Tools

- DE
- Open Zeppelin
- Code Analyzer
- Solidity Code
- Complier
- Hardhat

FUNCTION OVERVIEW

Can Take Back Ownership	Not Detected
Owner Change Balance	Not Detected
Blacklist	Not Detected
Modify Fees	Not Detected
Proxy	Not Detected
Whitelisted	Not Detected
Anti Whale	Not Detected
Trading Cooldown	Not Detected
Transfer Pausable	Not Detected
Cannot Sell All	Not Detected
Hidden Owner	Not Detected
Mint	Not Detected

VULNERABILITY CHECKLIST

Design Logic	Passed
Compiler warnings.	Passed
Private user data leaks	Passed
Timestamp dependence	Passed
Integer overflow and underflow	Passed
Race conditions & reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious Event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zeppelin module	Passed
Fallback function security	Passed

RISK CLASSIFICATION

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and access control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Low Risk

Issues on this level are minor details and warning that can remain unfixed.

Informational

Information level is to offer suggestions for improvement of efficacy or security for features with a risk free factor.

AUDIT SUMMARY

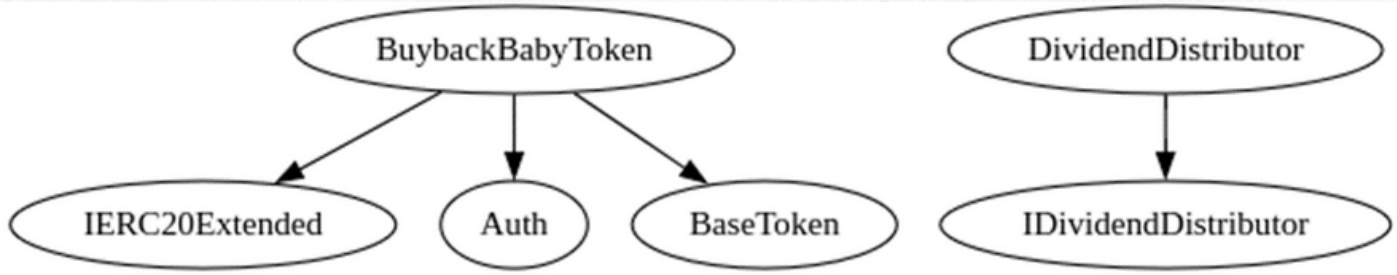
Used Tools:

1.Manual Review: The code has undergone a line-by-line review by the Expelee team.

2.BSC Test Network: All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.

3.Slither: The code has undergone static analysis using Slither.

Inheritance Trees:



Summary:

-Owner is not able to set fees over 25% (buy/sell/transfers)
- Owner is not able to set max buy/sell/transfer amounts
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to disable trades
- Owner is not able to mint new tokens

Functional Tests

Router (PCS V2):
0xD99D1c33F9fC3444f8101754aBC46c52416550D1

1- Adding liquidity (passed):
https://testnet.bscscan.com/tx/0xec74129b54ad3b9322128177b3d79e1e7152038115227edc9013ef1bd6f65aab

2- Buying when excluded from fees (up to 25% tax) (passed):
https://testnet.bscscan.com/tx/0xa7d9a808ec36d721fefbb1eb207064353dd18de5d0562d457b26d1e32277bf89

3- Selling when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0xa4aa4056e0a0018fd116ae81d4c62dba71498ecec678039647ac8cebf3191f8a

4- Transferring when not excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x6728dd2859344677a8bc973ab626fd0843988fa96d2c886e6638031bcc28cdea

5- Buying when not excluded from fees (up to 25% tax) (passed):
https://testnet.bscscan.com/tx/0x351fadd6c404549e385ff92ce7c6a134a9266cc64a089da62ef81d3eb36085c0

6- Selling when not excluded from fees (up to 25% tax) (passed):
https://testnet.bscscan.com/tx/0xe939b193281192ba970970634e85d44ebd93ec22f2db701108bae6fd45eede93

Functional Tests

7- Transferring when not excluded from fees (up to 25% tax) (passed):
https://testnet.bscscan.com/tx/0x76b706ef4b8bb7fe8f40def8b4ff3ac8c0693521771b923a0c716126250517e0

8- Internal swap (passed):
marketing wallet received ETH
https://testnet.bscscan.com/address/0x9d54c86d4b34ef6a2dc6352235f4e0f67f501ecf#internaltx

9- Auto buybacks (passed):
auto-buybacks happened after reaching the buyback threshold, auto-buyback can be seen here (dead wallet received tokens)
https://testnet.bscscan.com/tx/0x76b706ef4b8bb7fe8f40def8b4ff3ac8c0693521771b923a0c716126250517e0

10- Reflections (passed):
as seen in this transaction reward token is distributed to all holders
https://testnet.bscscan.com/tx/0x22306ea00f9f87f27903714b53b03aeecc0e8d1203cb5aaac75a2b8a52d7f1f1

11- Zeus buyback (passed):
https://testnet.bscscan.com/tx/0x969e0383303febc6c2bb4b7d0fe2f284542c27ab8e65a2bdc4fcfc4d8fce1fcb

MANUAL AUDIT

Severity Criteria

Expelee assesses the severity of disclosed vulnerabilities according to a methodology based on OWASP standards.

Vulnerabilities are divided into three primary risk categories: high, medium, and low.

High-level considerations for vulnerabilities span the following key areas when conducting assessments:

- Malicious Input Handling
- Escalation of privileges
- Arithmetic
- Gas use

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

FINDINGS

- High Risk Findings:1
- Medium Risk Findings:1
- Low Risk Findings:1
- Suggestions & discussion: 1
- Gas Optimizations : 1

Medium Risk Findings

Issues:
Owners ability to change fee
Type:
Centralization
Function:
_setFees
Line:
1653-1667
Overview:
Owner is able to set up to 25% fee on buy/sell/transfers:

```
function _setFees(
    uint256 _liquidityFee,
    uint256 _buybackFee,
    uint256 _reflectionFee,
    uint256 _marketingFee,
    uint256 _feeDenominator
) internal {
    liquidityFee = _liquidityFee;
    buybackFee = _buybackFee;
    reflectionFee = _reflectionFee;
    marketingFee = _marketingFee;
    totalFee = _liquidityFee.add(_buybackFee).add(_reflectionFee).add(
        _marketingFee
    );
    feeDenominator = _feeDenominator;
    require(
        totalFee <= feeDenominator / 4,
        "Total fee should not be greater than 1/4 of fee denominator"
    );
}
```

Recommendation
•Set proper safeguards to prevent high fees
•Increase transparency and give proper information about tax structure to community

Suggestions

Issue:
redundant code
Type:
code smell
Function:
onlyBuybacker
Line:
1237-1240
Severity:
Informational
Overview:
onlyBuybacker modifier never been used in the contract

```
modifier onlyBuybacker() {
    require(buyBacker[msg.sender] == true, "Not a buybacker");
    _;
}
```

ABOUT EXPELEE

Expelee is a product-based aspirational Web3 Start-up. Coping up with numerous solutions for blockchain Security and constructing a Web3 Ecosystem from Deal making platform to developer hosting open platform, while also developing our own commercial and sustainable blockchain.

www.expelee.com

expeleeofficial

expelee

Expelee

expelee

expelee_official

expelee-co

expelee

Building the Futuristic **Blockchain Ecosystem**

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 Expelee 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 Expelee 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 Expelee team disclaims any liability for the resulting losses.