expelee

Building the Futuristic Blockchain Ecosystem

SECURITY AUDIT REPORT

DISPE

TOKEN OVERVIEW

Risk Findings

Severity	Found
🔴 High	1
🟠 Medium	0
🟡 Low	0
🔵 Informational	0

Centralization Risks

Owner Privileges	Description
🟢 Can Owner Set Taxes >25% ?	Not Detected
🔴 Owner needs to enable trading ?	Yes, owner needs to enable trading
🟢 Can Owner Disable Trades ?	Not Detected
🟢 Can Owner Mint ?	Not Detected
🟢 Can Owner Blacklist ?	Not Detected
🟢 Can Owner set Max Wallet amount ?	Not Detected
🟢 Can Owner Set Max TX amount ?	Not Detected

TABLE OF CONTENTS

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	**-**
Audit Date	**7 July 2023**

CONTRACT DETAILS

Token Name: Disappointed Pepe

Symbol: DISPE

Network: Ethereum

Language: Solidity

Contract Address:
0xe4F9B2C98830D27333C400b24E725F8ad69aA35E

Total Supply: 420,000,000,000

Owner's Wallet:
0xaD67Df12c9Fc80C97042de68d7E660F9Cd857E0d

Deployer's Wallet:
0xaD67Df12c9Fc80C97042de68d7E660F9Cd857E0d

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
- Compiler
- Hardhat

expelee

VULNERABILITY CHECKS

Design Logic	Passed
Compiler warnings	Passed
Private user data leaks	Passed
Timestamps 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 Zepplin module	Passed

RISK CLASSIFICATION

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and acces 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.
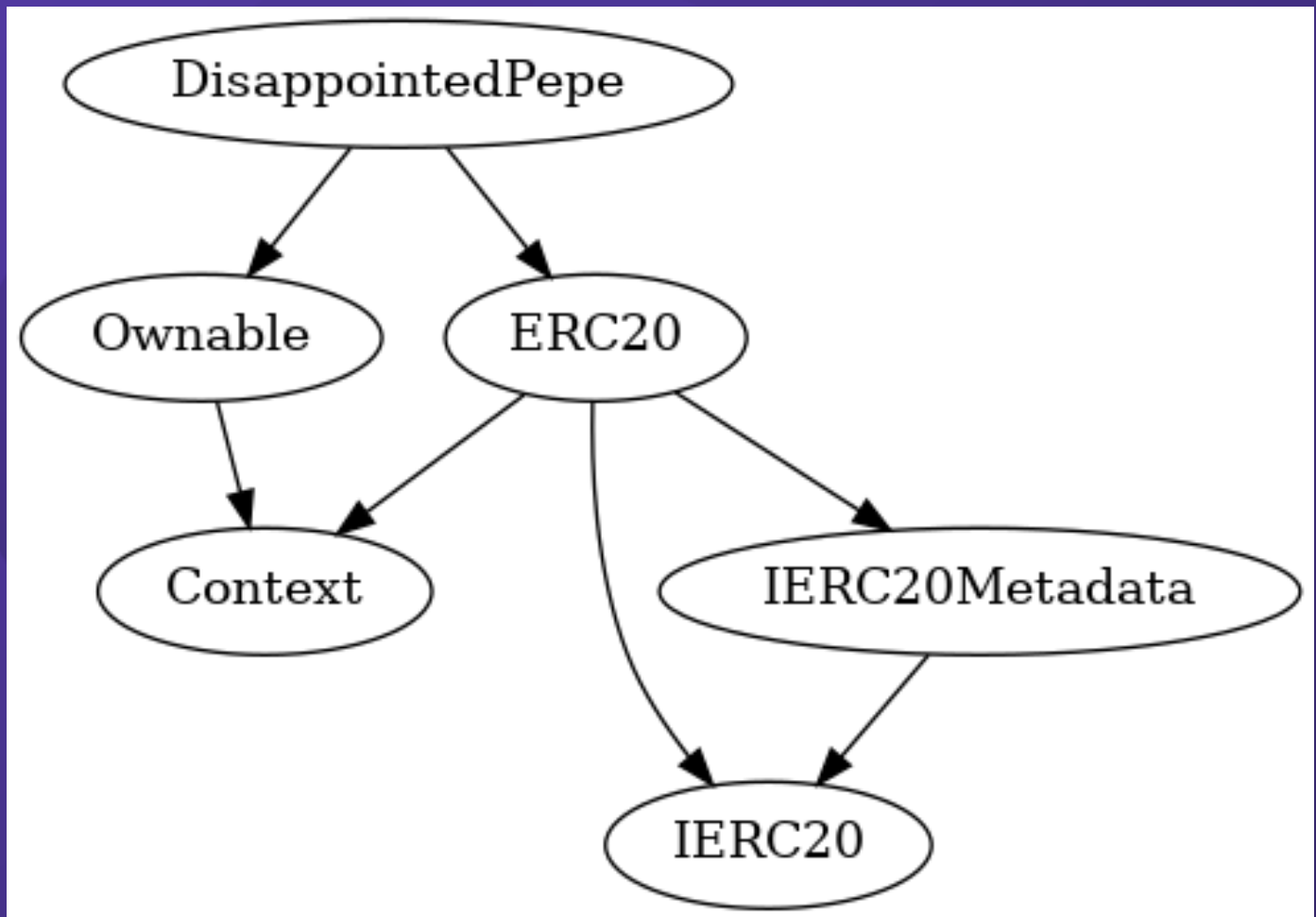
Low Risk

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

Informational

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

INHERITANCE TREES

TESTNET VERSION

Adding Liquidity ✅
Tx:
https://testnet.bscscan.com/tx/0xae682cb20f36dc86ad2ec
55e158df7baaee7e77f1ee3b89f94a9160ea6902eeb

===

Buying ✅
Tx (0% tax):
https://testnet.bscscan.com/tx/0x5ab6d157e144a3e3610643
b6ac90c7654497852a0fadb3e75d4e9b1c452fdc9a

===

Selling ✅
Tx (0% tax):
https://testnet.bscscan.com/tx/0xa8a66f846eeb256abee9b
009042d0a5795ed997bac475fa10f229354c6b27ed6

===

Transferring ✅
Tx (0% tax):
https://testnet.bscscan.com/tx/0x5fd73b0ddc054a37ce427
58f85ced4764dc0008f4e13b0dc36f230dbbb3c292b

===

FUNCTION DETAILS

```
| Contract |      Type      |    Bases    |       |        |
|:---------:|:--------------:|:------------:|:-------------:|:-------------:|
|     └     | **Function Name** | **Visibility** | **Mutability** | **Modifiers** |
||||||
| **Context** | Implementation | |||
| └ | _msgSender | Internal 🔒 | ||
| └ | _msgData | Internal 🔒 | ||
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗ | 🛑 |NO❗ |
| └ | owner | Public ❗ | |NO❗ |
| └ | renounceOwnership | Public ❗ | 🛑 | onlyOwner |
| └ | transferOwnership | Public ❗ | 🔴 | onlyOwner |
| └ | _transferOwnership | Internal 🔒 | 🛑 ||
||||||
| **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❗ |
||||||
```

FUNCTION DETAILS

```
| **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 🔒 | 🔴 ||
| └ | _approve | Internal 🔒 | 🔴 ||
| └ | _beforeTokenTransfer | Internal 🔒 | 🔴 ||
| └ | _afterTokenTransfer | Internal 🔒 | 🔴 ||
||||||
| **DisappointedPepe** | Implementation | Ownable, ERC20 |||
| └ | <Constructor> | Public ❗ | 🔴 | ERC20 |
| └ | isAuthorized | Public ❗ | |NO ❗ |
| └ | authorize | Public ❗ | 🔴 | onlyOwner |
| └ | unauthorize | Public ❗ | 🔴 | onlyOwner |
| └ | setTrading | External ❗ | 🔴 | onlyOwner |
| └ | _beforeTokenTransfer | Internal 🔒 | 🔴 ||
| └ | burn | External ❗ | 🔴 |NO ❗ |
```

Legend

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

MANUAL REVIEW

Severity Criteria

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

Vulnerabilities are dividend into three primary risk categroies:
High
Medium
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		

HIGH RISK FINDING

Category: **Centralization**
Subject: Trades are disabled by default
Status: Open
Impact: **High**
Overview:
Owner of the contract must enable trades manually for holders. If trades remain disabled, holders won't be able to trade their tokens.

```
function setTrading() external onlyOwner {
    require(tradingActive == false);
    tradingActive = true;
}
```

Suggestion:
Its suggested to enable trades before presale which can be achieved by calling "setTrading" functino or transfer ownership of the contract to a trusted 3rd party e.g. a certified safu developer.

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　　in expelee

📷 expelee_official　　🐙 expelee-co

exp̂elee

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 guarantess against the sale of team tokens or the removal of liquidity by the project audited in this document.

Always do your own research and project 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. Alway 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.

exⱒelee

Building the Futuristic Blockchain Ecosystem