expelee

Building the Futuristic **Blockchain Ecosystem**

Audit Report

FOR

Eraora Group

OVERVIEW

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	Done
Audit Date	17 Sep 2022

Why Passed?

ERAORA token is a simple ERC20 token with no additional functionalities that need to be audited, all of the used libraries & contracts are safe and audited by openzeppelin.

- Team Expelee

PROJECT DESCRIPTION

Eraora Group

ERAORA project aims to offer the international community a new way to buy safely.
A reliable, user-friendly and convenient alternative to buy products with app and metaverse and the certainty of their originality and provenance.
The first project to work with companies around the world & distribute revenue to token holders & Stakers.

eraora.io

eraoraio

eraorastore

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

- Team Expelee

CONTRACT DETAILS

Contract Name
ERC 20

Optimization
Yes with 200 runs

Contract Address (Verified)
0xbb104D22ba9EcB107e77B83c58AD9F8aF230284F

Network
ERC 20

Language
Solidity

Total Supply
9,000,000,000 $EOT

Decimals
18

Compiler
v0.8.17+commit.8df45f5f

License
MIT license

Contract SHA-256 Chechsum:

0fb3633eb3c5d8a2c30cad23f7ddba68b6ed1aa163e88cfed06e3b3406aee364

What is checksum?
This is the hash signature of contract source code, if anything even a tiny word changes in the contract this signature would betotaly different, use it to know if the team is using the same contract that we audited or not.

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
Creator Address	0x999619733a2df505229967db54b84d6f4709dc45
Creator Balance	3240000000 $EOT
Owner Address	0x35096d07c0fa7a8bb68353f5275913c381882f22
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

Slither, Echidna, etc - we used automated static-analysis tools to check contract for common solidity vulnerability & mistakes.

UniswapV2 Fork - We launched **ERAORA** token on our Local Blockchain (Hardhat) & we performed couple of buys & sells & transfers to make sure that there won't be any problem regarding the trades.

Manual Review:
we spent most of the audit process time reading the whole contract line by line, we even checked standard libraries & contracts (ERC20, Safemath, etc).

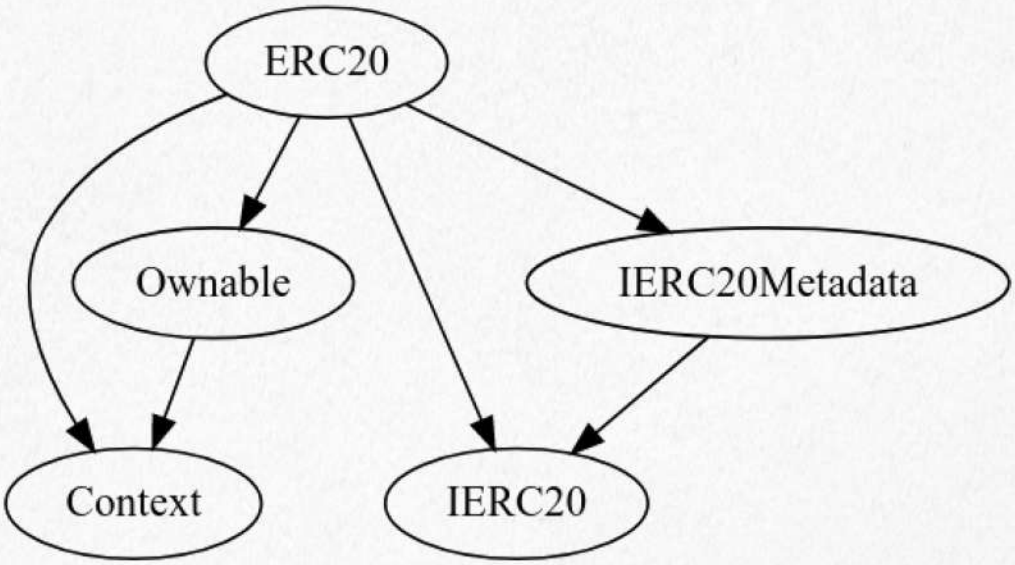
Ownership & Owner privileges:

Although there is not any type of centralization risks or onlyOwner functions inside the contract and owner privileges over contract is not a concern, but current owner of the contract is: **0x35096d07c0fa7a8bb68353f5275913c381882f22** the only concern (at time of writing this audit report) is that owner is holding 36% of total supply.

Contracts & Inheritance Tree:

All of below contracts are in this audit scope

- ERC20.sol
- ERC20.sol
- ERC20Metadata
- Ownable
- Context



Local Blockchain Launch Test:

We added 10% of total supply with 200BNB to the uniswap pool, we performed 10 stimulated buys and 10 stimulated sells, all of them were successful with no errors, there was no tax, no overflows, no uniswap errors, no high gas issues.

Buys:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

yarn run v1.22.15
warning ../../../package.json: No license field
$ hh run utils/Actions/buy.js --network localhost
🟢Action : Buy
Trying to buy : 2237672.601041903056631346 EOT
 🔸Buyer : 0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC
 🔸Received Amount : 2237672.601041903056631346 EOT
 🔸Buy Tax : 0 %
 🔸Gas Used : 153567
 🔸Monitored Wallets :
Contract : 0.0 EOT
==================================
🟢Action : Buy
Trying to buy : 2226556.517627647028839863 EOT
 🔸Buyer : 0x90F79bf6EB2c4f870365E785982E1f101E93b906
 🔸Received Amount : 2226556.517627647028839863 EOT
 🔸Buy Tax : 0 %
 🔸Gas Used : 119391
 🔸Monitored Wallets :
Contract : 0.0 EOT
==================================
🟢Action : Buy
Trying to buy : 2215523.102079386344227835 EOT
 🔸Buyer : 0x15d34AAf54267DB7D7c367839AAf71A00a2C6A65
 🔸Received Amount : 2215523.102079386344227835 EOT
 🔸Buy Tax : 0 %
 🔸Gas Used : 119391
 🔸Monitored Wallets :
Contract : 0.0 EOT
==================================
undefined
Done in 5.37s.
```

Sells:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

$ hh run utils/Actions/sell.js --network localhost
0xf2AB1abFdB630e88c46197473EF260C9f9E016Af
🟠Action : Sell
 🔸Seller : 0x3C44CdDdB6a900fa2b585dd299e03d12FA4293BC
 🔸Sell Amount : 900000.0
 🔸Sell Tax : 0 %
 🔸Gas Used : 118014
 🔸Block Number : 17
 🔸Monitoring Wallets:
Contract : 0.0 ERAORA
==================================
🟠Action : Sell
 🔸Seller : 0x90F79bf6EB2c4f870365E785982E1f101E93b906
 🔸Sell Amount : 900000.0
 🔸Sell Tax : 0 %
 🔸Gas Used : 118026
 🔸Block Number : 19
 🔸Monitoring Wallets:
Contract : 0.0 ERAORA
==================================
🟠Action : Sell
 🔸Seller : 0x15d34AAf54267DB7D7c367839AAf71A00a2C6A65
 🔸Sell Amount : 900000.0
 🔸Sell Tax : 0 %
 🔸Gas Used : 118026
 🔸Block Number : 21
 🔸Monitoring Wallets:
Contract : 0.0 ERAORA
==================================
undefined
Done in 4.81s.
```

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 Summary

- **High Risk Findings:** 0
- **Medium Risk Findings:** 0
- **Low Risk Findings:** 0
- **Suggestions & Discussion:** 2

Suggestions & Discussion

[S-1] - Change contract name from **ERC20** to another name close to token name, like **ERAORA**, thiscaused some compilation issues (not important however) in our testing environment, because there was a collision between the contract name & openzeppelin ERC20 contract.

[S-2] - emit an event from **claimStuckTokens** function, transfering BNB tokens doesn't emit anyevents.

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.