expelee

Building the Futuristic Blockchain Ecosystem

Security Audit Report

FOR

Orbiter Token

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 with Critical Risk
KYC Verification	Not Done
Audit Date	20 March 2023

PROJECT DESCRIPTION

Orbiter Token

ORBITER Farm is a decentralized finance (DeFi) project based on Arbitrum One. It aims to provide a platform for users to earn passive income through staking and liquidity provision. By staking their assets in ORBITER Farm's liquidity pools, users can earn rewards in the form of the platform's native token, $ORB. Additionally, the platform features a yield farming mechanism that allows users to maximize their returns through strategic placement of their assets in the highest yielding pools. The project's underlying technology is built on blockchain, ensuring security and transparency for all participants in the ecosystem.

Note

The token has a mint function that can only be called by the owner. The ownership has been transferred to a farming contract (similar to the Sushi Master Chef contract), which allows minting new tokens. However, we have not audited the farming contract, and therefore, we cannot guarantee its security or verify whether there are any potential exploits that could be used to mint new tokens indirectly. This poses a significant risk as it might lead to uncontrolled token issuance or other unforeseen consequences. The token does not have a specified maximum supply. Without a cap on the total number of tokens that can be minted, there is a risk of inflation or token value dilution over time. This could negatively impact the token's value and the overall confidence of investors and users in the ecosystem.

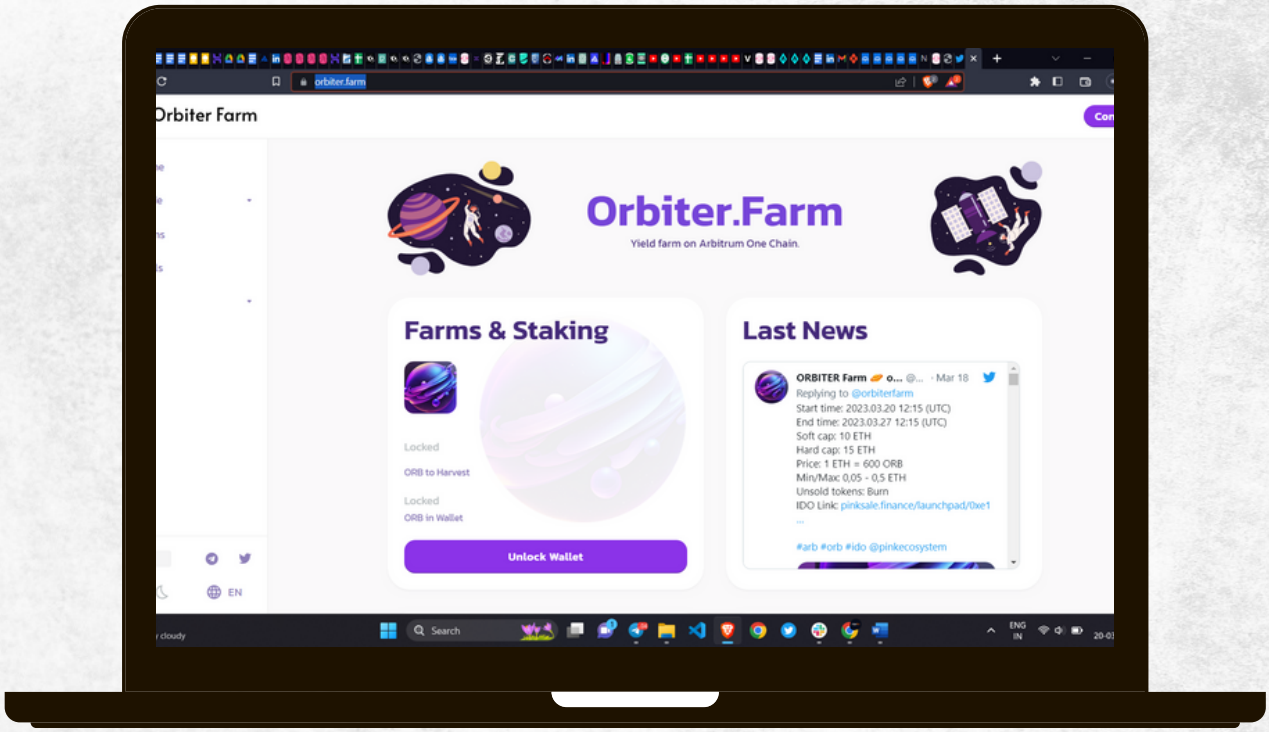
we highly recommend proceeding with caution and considering an audit of the farming contract before any further steps are taken. Addressing these concerns is vital for ensuring the token's security, stability, and long-term success.

expelee

expelee.com

Social Media Profiles

Orbiter Token



https://orbiter.farm/

https://t.me/orbiterfarm

https://twitter.com/orbiterfarm

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

-Team Expelee

CONTRACT DETAILS

Token Name
Orbiter Token

Symbol
ORB

Network
Arbitrum

Language
Solidity

Contract Address (Verified)
0x126AB8350e3172b349785CDdf53851329AC344be

Token Type
BEP20

Total Supply
20,000 (initial supply)

Contract SHA-256 Checksum:
97a5b17b0fd90ece89930aeff76cc32fef1a6f14

Owner Wallet
0x1ba62EB5994D562cb41BE2dbfeDDc141177D30D6

Deployer Wallet
0xd379c43A84E4db559ce2a433D625106966718F8a

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	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

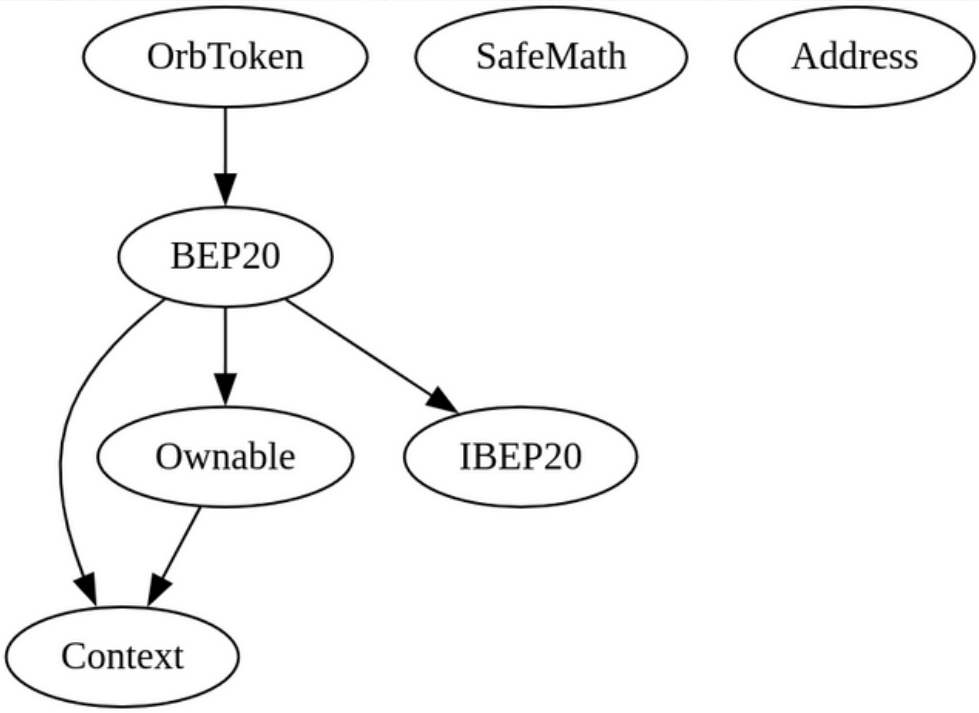
Used Tools:

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

2.Goerli Test Network: All tests were conducted on the Goerli 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 buy/sell/transfer fees
- Owner is not able to set max buy/sell/transfer amounts or any other limitations
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to disable trades
- There is a mint function in the contract, but, owner is expected to not be able to mint new tokens since contract ownership is transferred to another contract. however, we are not provided with security audit or details of that contract

Functional Tests

1- Adding liquidity (passed):
https://testnet.bscscan.com/tx/0xac94a790db8e07253677d47ccf1670db6eba6d882a82bc093785a2e6de33c038

2- Buying (0% tax) (passed):
https://testnet.bscscan.com/tx/0x7a1ba9ac86bbfef192834fd24196c5fc1e5760743672725638e0f33581b76894

3- Selling (0% tax) (0% tax) (passed):
https://testnet.bscscan.com/tx/0x3362780d21344cd976184653f8d6217535d20fd3b97fed9033ed8e4fd10ca449

4- Transferring (0% tax) (0% tax) (passed):
https://testnet.bscscan.com/tx/0xb57341c5e2a87faeabbead61cc6e3bf10d5afeeffc86100f8bbec8aee3b82d82

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

- Critical Risk Findings:1
- High Risk Findings:0
- Medium Risk Findings:0
- Low Risk Findings:0
- Suggestions & discussion: 1
- Gas Optimizations : 0

Critical Risk Findings

Centralization - Minting function

Severity: Critical
Function: mint
Lines: 792 and 905
Status: not resolved
Overview:

The Mint functions allow owner to create unlimited new tokens (no max supply), however, the contract is owned by a farming contract:

https://arbiscan.io/address/0x1ba62eb5994d562cb41be2dbfeddc141177d30d6#code

and as indicated by owner of the token in the documentation, only the farming contract is able to mint new tokens.

Since the farming contract is not in this audit scope, we do not know if owner is able to exploit or abuse farming contract functionality to somehow mint new tokens specially since farming contracts often has a mechanism to give owner this ability to update token's emission rate per block, which if is set to a very high value can be used to exploit the project.

```
function mint(address _to, uint256 _amount) public onlyOwner {
    _mint(_to, _amount);
    _moveDelegates(address(0), _delegates[_to], _amount);
}
```

Critical Risk Findings

Recommendation:

•Define a reasonable max supply to now allow unlimited mints
•Perform or provide a security audit of farming contract by a trusted and valid audit firm to mitigate above issue
•Increase transparency by declaring the minting plans and structure
•Create a DAO governance model to allow holders decide the emission rate or other minting related events
•Delete minting function and instead use a reasonable fee plan to reward stakers at farming contract
the project.

Suggestions & Discussion

Informative – 2 Mint function

Severity: low
Function: mint
Lines: 792 and 905
Status: not resolved
Overview:

Currently, there are 2 separate mint functions in the contract, one that only mints to msg.sender, and the other one that mints to a specific address and also mints voting power for the receiver.
Using 2 mint functions with same name can create confusion for investors and increases the chance of mistake if used in another contract like farming.

```
function mint(address _to, uint256 _amount) public onlyOwner {
    _mint(_to, _amount);
    _moveDelegates(address(0), _delegates[_to], _amount);
}
```

Recommendation:

•Define different name for each one

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.