expelee

Building the Futuristic Blockchain Ecosystem

SECURITY AUDIT REPORT

OracleAi

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:

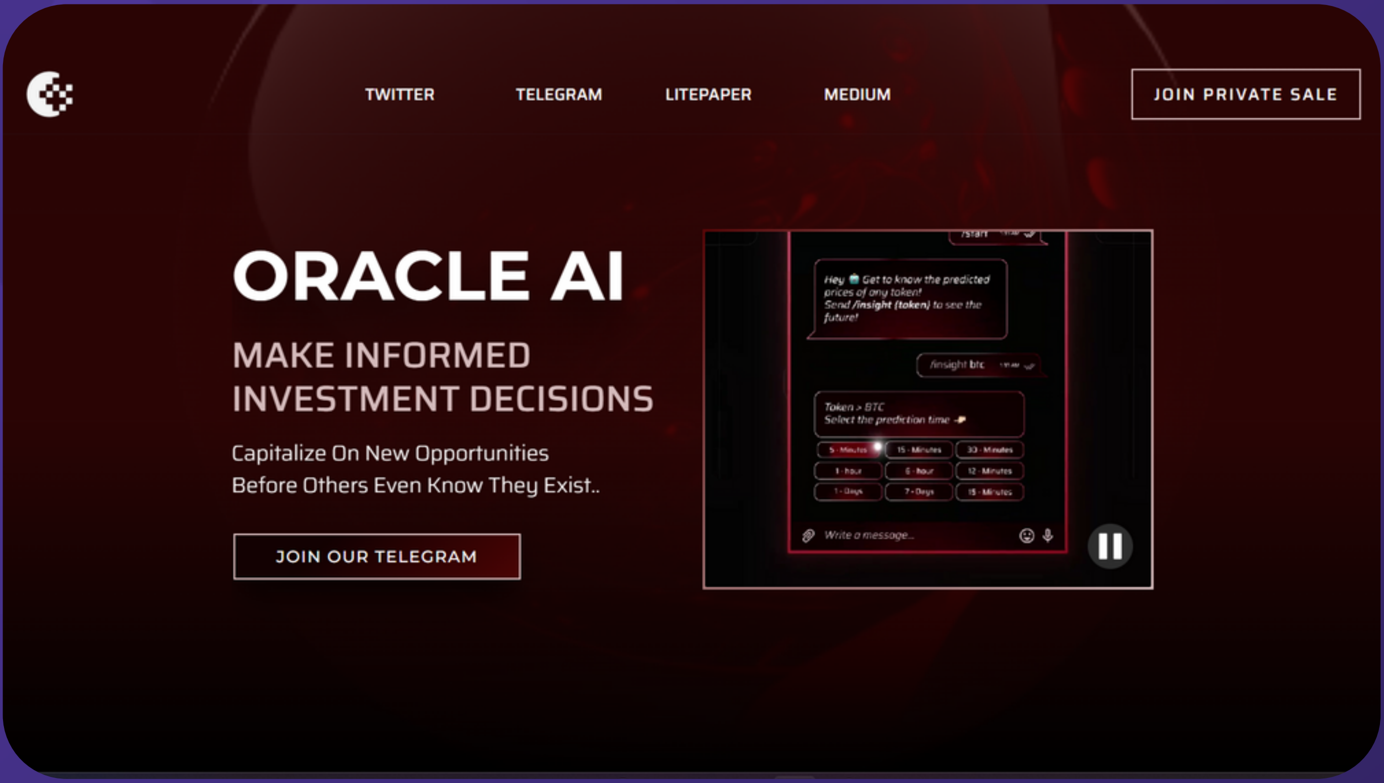
Audit Result	**Passed**
KYC Verification	**Done**
Audit Date	**9 May 2023**

PROJECT DESCRIPTION

The Oracle AI predictive model analyzes historical and current market data, along with a range of other factors such as economic indicators, company financials, and industry trends, resulting in highly accurate predictions for asset valuations and market trends.

SOCIAL MEDIA PROFILES

OracleAi



ORACLE AI

MAKE INFORMED INVESTMENT DECISIONS

Capitalize On New Opportunities Before Others Even Know They Exist..

/start

Hey Get to know the predicted prices of any token! Send **/insight (token)** to see the future!

/insight btc

Token > BTC
Select the prediction time

https://t.me/oracleaiofficial

https://twitter.com/orcai_

https://www.orcai.io/

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

Team Expelee

CONTRACT DETAILS

Token Name: Oracle AI

Symbol: ORCAI

Network: Arbitrum Chain

Language: Solidity

Contract Address: with local files

Total Supply: 1000000

Contract SHA-256 Checksum:

Owner's Wallet:

Deployer's Wallet:

OWNER PRIVILEGES

- Owner can pause trade
- Owner can set redis fees up to 100%
- Owner can set the max tx amount "0"
- Owner can set the max wallet amount "0"
- Owner can exclude accounts from fees
- Owner can set taxFees 20% at max
- Owner can set SwapTokensThreshold without limit
- Owner can change swap settings
- Owner can't add an account to bot list after deploy but can add some address in the constructor.

Important Note:

uint256 public _maxTxAmount = 100000000;
_maxTxAmount is set to 100000000 this is equal to just 100 token.

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

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.

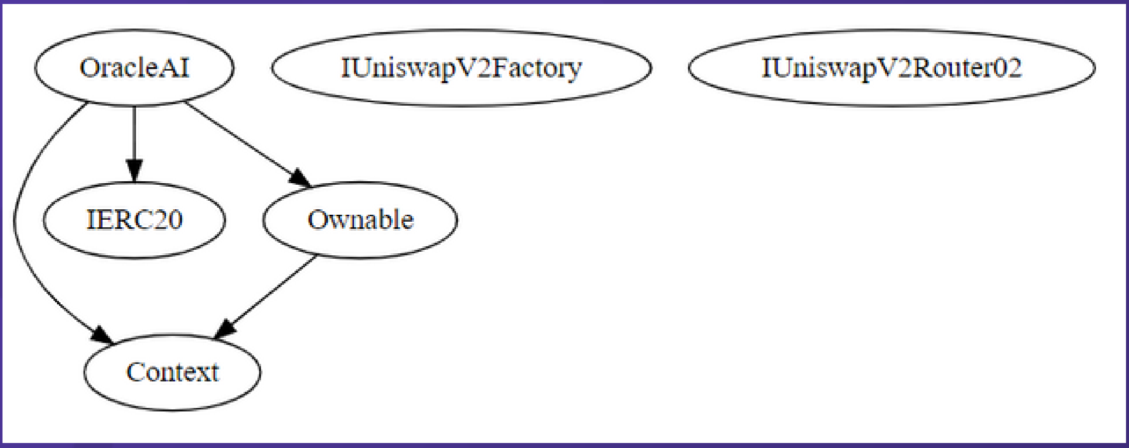
Low Risk

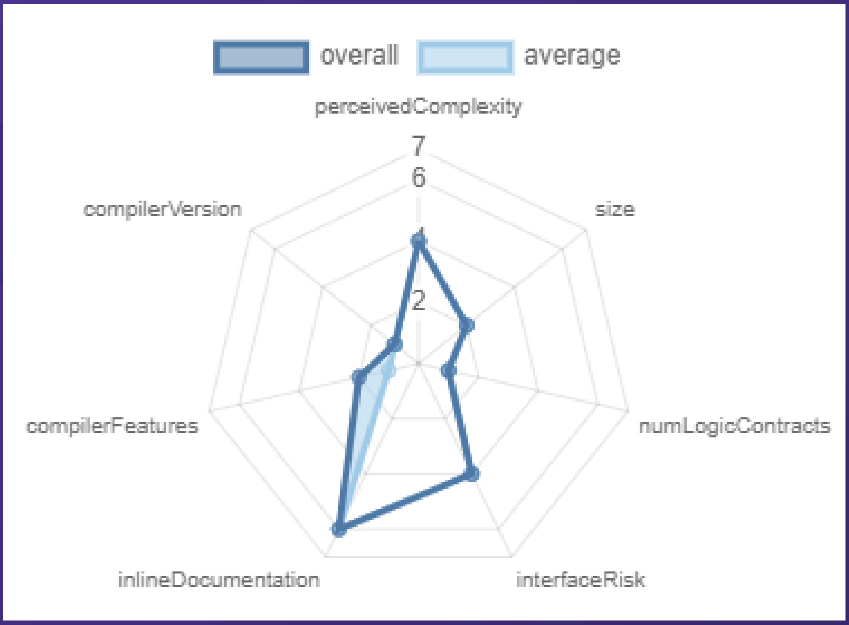
Issues on this level are minor details and warning that can remain unfixed.

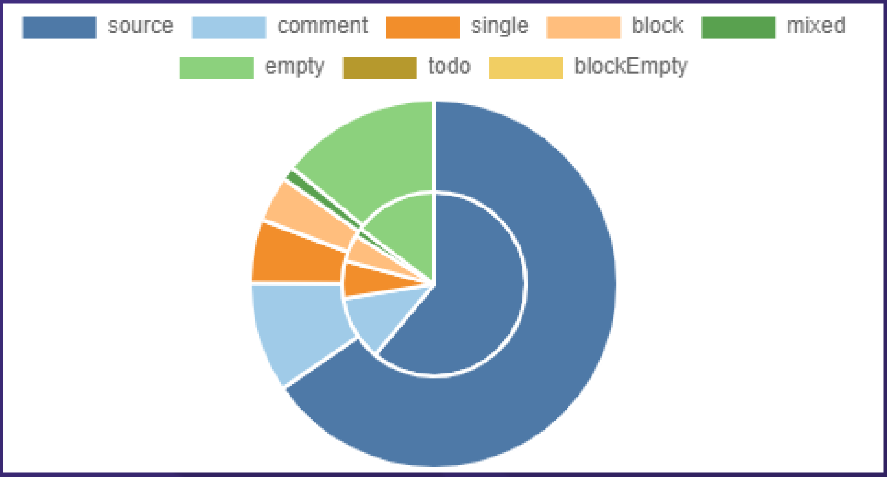
Informational

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

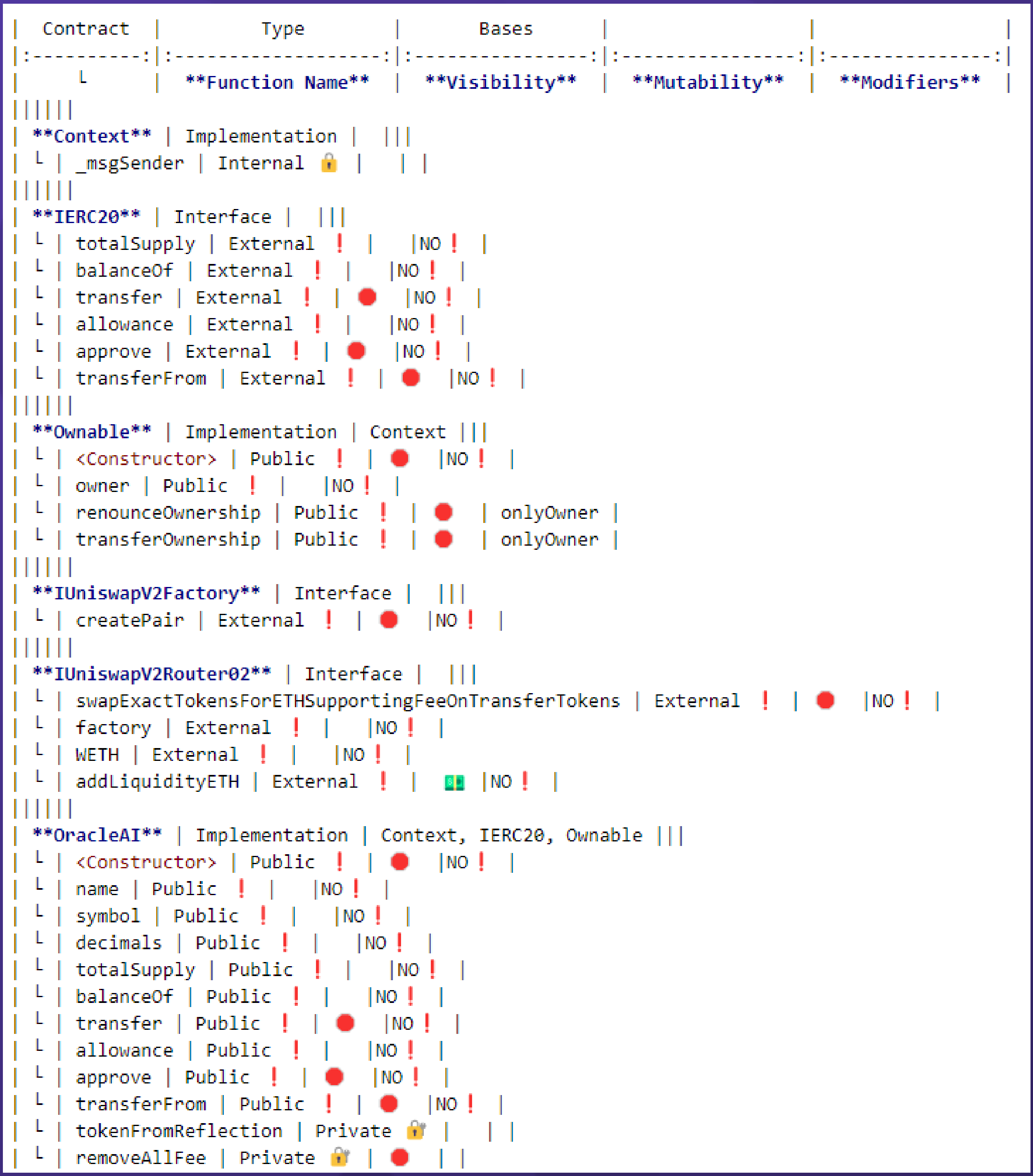
INHERITANCE TREES





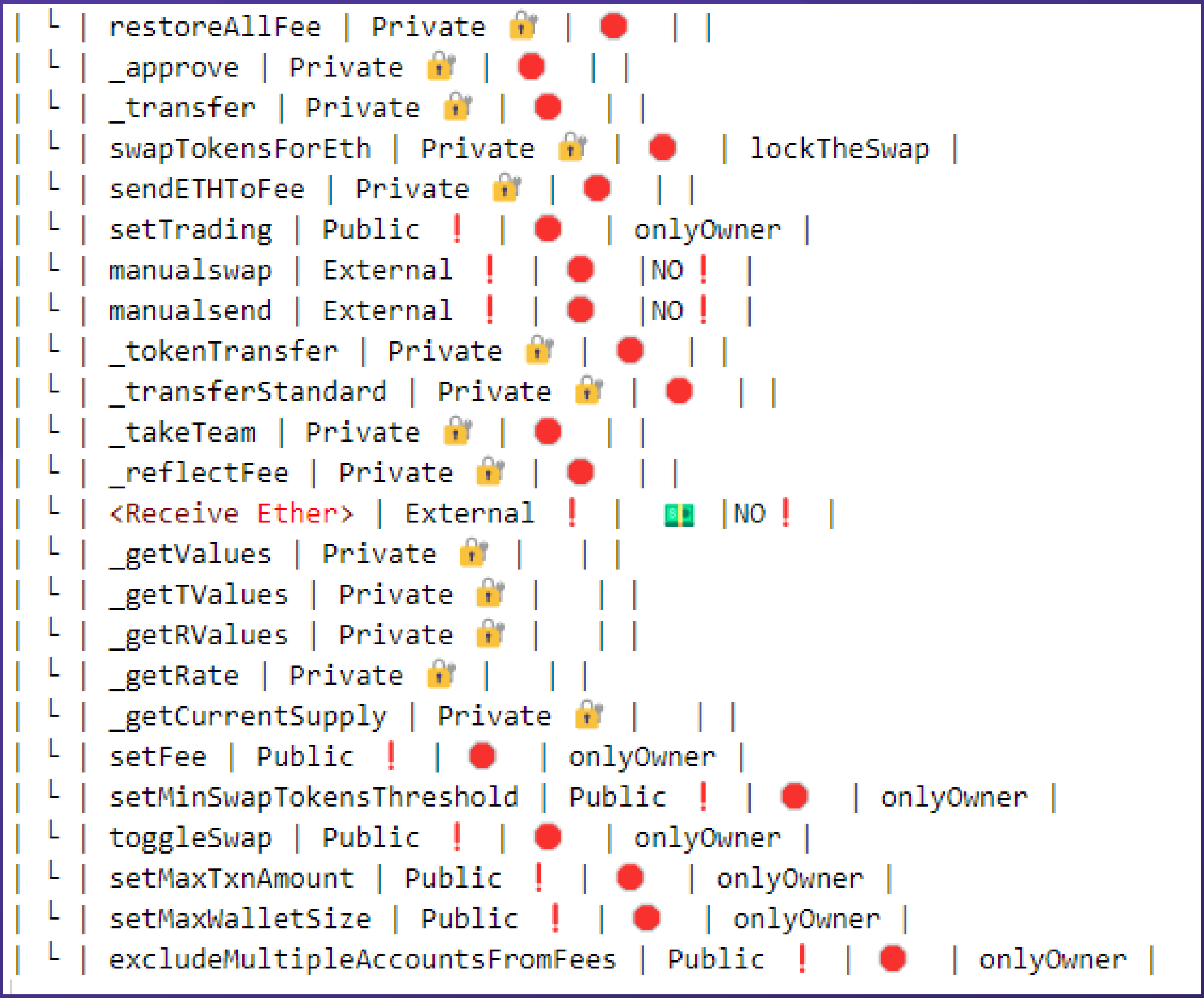


FUNCTION DETAILS



```
| Contract  |        Type        |       Bases      |                  |                 |
|:----------:|:-------------------:|:----------------:|:----------------:|:---------------:|
|     └      |  **Function Name**  |  **Visibility**  |  **Mutability**  |  **Modifiers**  |
||||||
| **Context** | Implementation |  |||
| └ | _msgSender | Internal 🔒 |   | |
||||||
| **IERC20** | Interface |  |||
| └ | totalSupply | External ❗️ |   |NO❗️ |
| └ | balanceOf | External ❗️ |   |NO❗️ |
| └ | transfer | External ❗️ | 🛑  |NO❗️ |
| └ | allowance | External ❗️ |   |NO❗️ |
| └ | approve | External ❗️ | 🛑  |NO❗️ |
| └ | transferFrom | External ❗️ | 🛑  |NO❗️ |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗️ | 🛑  |NO❗️ |
| └ | owner | Public ❗️ |   |NO❗️ |
| └ | renounceOwnership | Public ❗️ | 🛑  | onlyOwner |
| └ | transferOwnership | Public ❗️ | 🛑  | onlyOwner |
||||||
| **IUniswapV2Factory** | Interface |  |||
| └ | createPair | External ❗️ | 🛑  |NO❗️ |
||||||
| **IUniswapV2Router02** | Interface |  |||
| └ | swapExactTokensForETHSupportingFeeOnTransferTokens | External ❗️ | 🛑  |NO❗️ |
| └ | factory | External ❗️ |   |NO❗️ |
| └ | WETH | External ❗️ |   |NO❗️ |
| └ | addLiquidityETH | External ❗️ |  💵 |NO❗️ |
||||||
| **OracleAI** | Implementation | Context, IERC20, Ownable |||
| └ | <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❗️ |
| └ | tokenFromReflection | Private 🔐 |   | |
| └ | removeAllFee | Private 🔐 | 🛑  | |
```

FUNCTION DETAILS



```
| └ | restoreAllFee | Private 🔐 | 🛑  | |
| └ | _approve | Private 🔐 | 🛑  | |
| └ | _transfer | Private 🔐 | 🛑  | |
| └ | swapTokensForEth | Private 🔐 | 🛑  | lockTheSwap |
| └ | sendETHToFee | Private 🔐 | 🛑  | |
| └ | setTrading | Public ❗ | 🛑  | onlyOwner |
| └ | manualswap | External ❗ | 🛑  |NO ❗ |
| └ | manualsend | External ❗ | 🛑  |NO ❗ |
| └ | _tokenTransfer | Private 🔐 | 🛑  | |
| └ | _transferStandard | Private 🔐 | 🛑  | |
| └ | _takeTeam | Private 🔐 | 🛑  | |
| └ | _reflectFee | Private 🔐 | 🛑  | |
| └ | <Receive Ether> | External ❗ |  💵 |NO ❗ |
| └ | _getValues | Private 🔐 |   | |
| └ | _getTValues | Private 🔐 |   | |
| └ | _getRValues | Private 🔐 |   | |
| └ | _getRate | Private 🔐 |   | |
| └ | _getCurrentSupply | Private 🔐 |   | |
| └ | setFee | Public ❗ | 🛑  | onlyOwner |
| └ | setMinSwapTokensThreshold | Public ❗ | 🛑  | onlyOwner |
| └ | toggleSwap | Public ❗ | 🛑  | onlyOwner |
| └ | setMaxTxnAmount | Public ❗ | 🛑  | onlyOwner |
| └ | setMaxWalletSize | Public ❗ | 🛑  | onlyOwner |
| └ | excludeMultipleAccountsFromFees | Public ❗ | 🛑  | onlyOwner |
```

MANUAL REVIEW

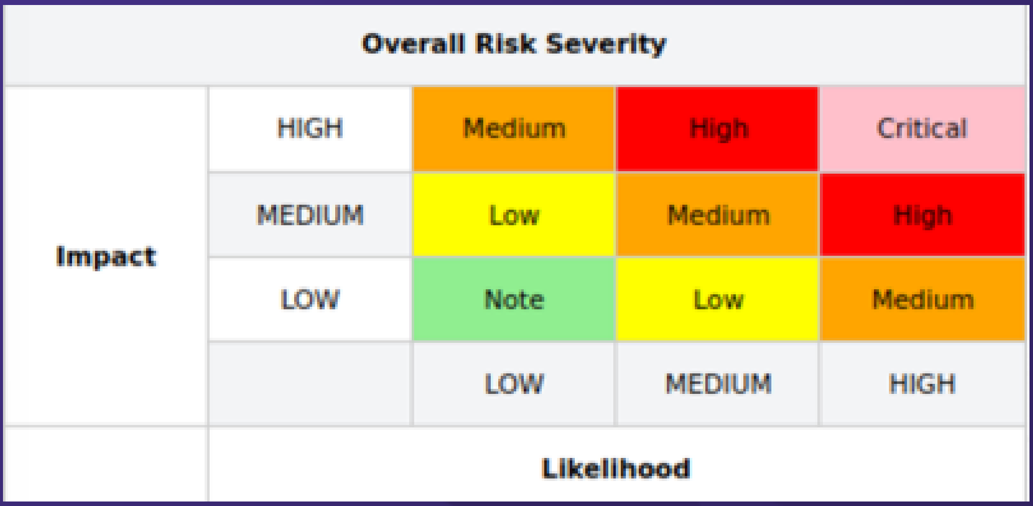
Severity Criteria

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			

FINDINGS

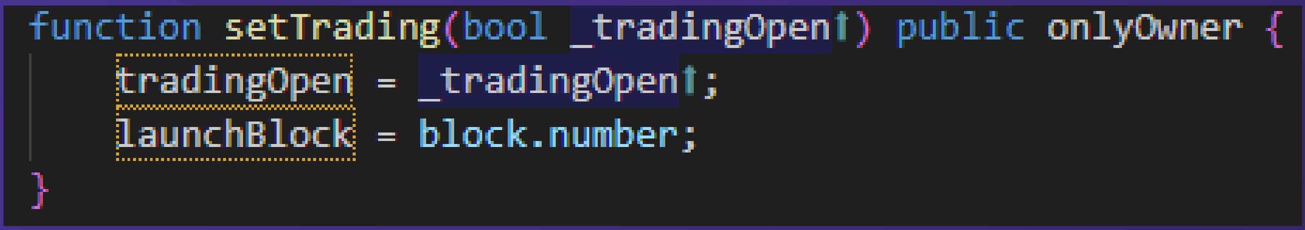
Findings	Severity	Found
High Risk	● High	4
Medium Risk	● Medium	0
Low Risk	● Low	4
Suggestion & discussion	● Informational	0
Gas Optimizations	● Gas Opt.	0

HIGH RISK FINDING

Owner can pause trade

Severity : High

Overview
The **setTrading** function is allows the contract operator to pause or unpause the contract. The function takes a boolean parameter **_tradingOpen** that determines whether the contract is paused or not. If the **_tradingOpen** parameter is set to false, the function sets Trading is stop.



```
function setTrading(bool _tradingOpen) public onlyOwner {
    tradingOpen = _tradingOpen;
    launchBlock = block.number;
}
```

Recommendation
The **setTrading** function, as it is currently implemented, presents a security risk as it allows the owner of the contract to completely halt trading. This can result in a situation where the owner may misuse their authority, intentionally or unintentionally, and disrupt the trading activities of the contract.

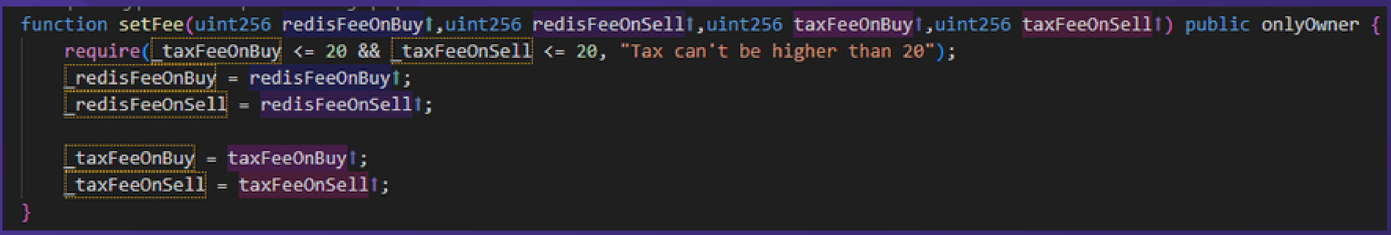
HIGH RISK FINDING

Owner can set redis fees up to 100%

Severity : High

Overview

The function checks whether the tax fee is higher than 20% and reverts if it is. However, there are no checks in place to prevent the owner from setting the Redis fees.



```
function setFee(uint256 redisFeeOnBuy,uint256 redisFeeOnSell,uint256 taxFeeOnBuy,uint256 taxFeeOnSell) public onlyOwner {
    require(_taxFeeOnBuy <= 20 && _taxFeeOnSell <= 20, "Tax can't be higher than 20");
    _redisFeeOnBuy = redisFeeOnBuy;
    _redisFeeOnSell = redisFeeOnSell;

    _taxFeeOnBuy = taxFeeOnBuy;
    _taxFeeOnSell = taxFeeOnSell;
}
```

Recommendation

The security of the **setFee** function can be improved, and the risk of malicious activity can be mitigated while still allowing the owner to set reasonable fees within the contract.

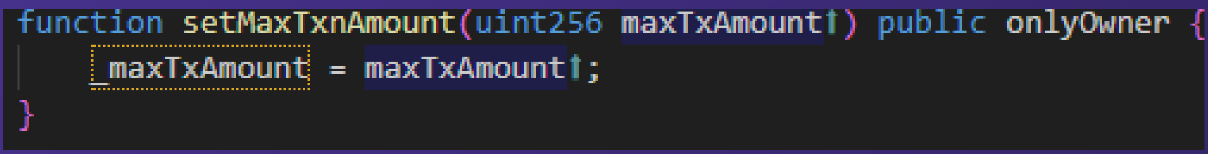
HIGH RISK FINDING

Owner can set the max tx amount "0"

Severity : HIGH

Overview

setMaxTxnAmount function allows the owner to set the maximum transaction amount for buying or selling tokens in single transaction. f the owner sets a "0" value to the maximum transaction amount, then it will not be possible for anyone to buy or sell more than that amount in a single transaction. This could potentially result in a situation where traders are unable to buy or sell enough tokens to meet their needs



```
function setMaxTxnAmount(uint256 maxTxAmount) public onlyOwner {
    _maxTxAmount = maxTxAmount;
}
```

Recommendation

It is recommended to set reasonable limits for the maximum transaction amount that balances the needs of users with the overall stability and security of the token

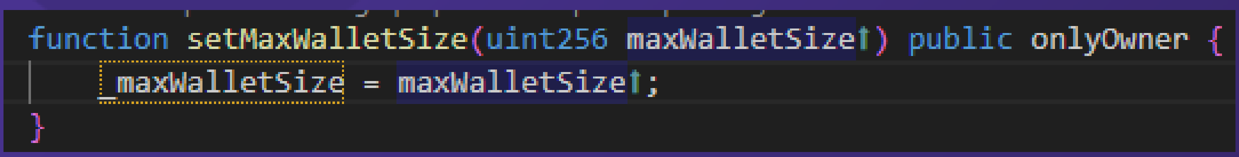
HIGH RISK FINDING

Owner can set the max wallet amount "0"

Severity : High

Overview

setMaxWalletSize function sets the maximum allowed balance that a wallet address can hold



```
function setMaxWalletSize(uint256 maxWalletSize) public onlyOwner {
    _maxWalletSize = maxWalletSize;
}
```

Recommendation

If the maximum wallet size is set too low, it may create barriers for legitimate token holders who wish to buy and hold the tokens for long periods of time. Additionally, if the maximum wallet size is set too high, it may increase the risk of price manipulation by large holders.

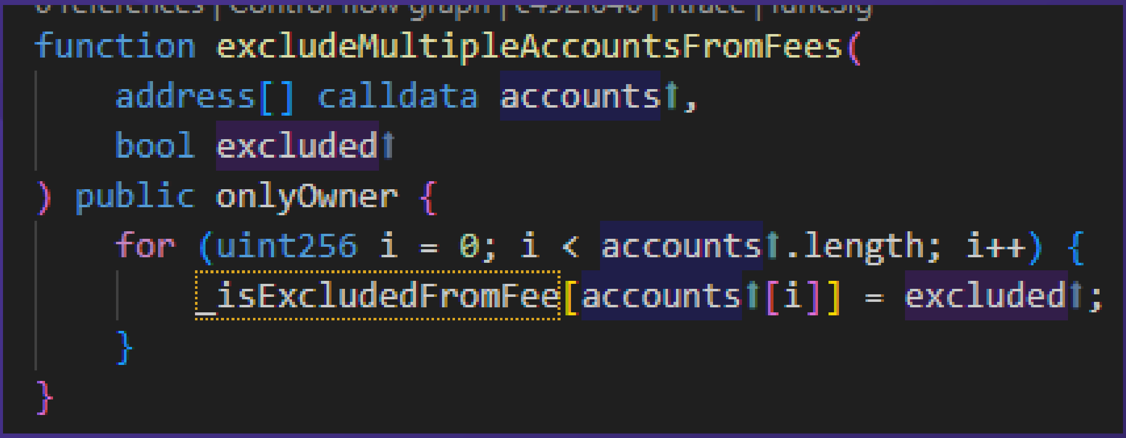
LOW RISK FINDING

Owner can exclude account from fee

Severity : Low

Overview

Excludes/Includes an address from the collection of fees



```
function excludeMultipleAccountsFromFees(
    address[] calldata accounts,
    bool excluded
) public onlyOwner {
    for (uint256 i = 0; i < accounts.length; i++) {
        _isExcludedFromFee[accounts[i]] = excluded;
    }
}
```

Recommendation

It is recommended to add additional access control measures, such as multi-factor authentication or time-based restrictions, to limit the number of authorized users who can call these functions. The contract owner account is well secured and only accessible by authorized parties.

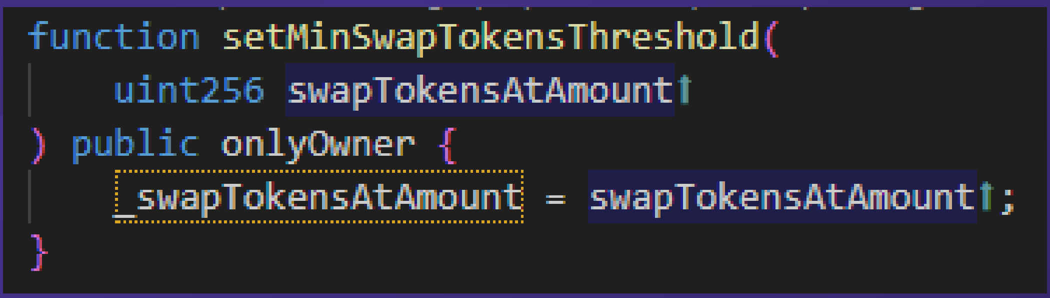
LOW RISK FINDING

Owner can change swapTokensAmount without limit

Severity : Low

Overview

setMinSwapTokensThreshold function allows the owner to set the minimum number of tokens required to trigger an automatic swap.



```
function setMinSwapTokensThreshold(
    uint256 swapTokensAtAmount
) public onlyOwner {
    _swapTokensAtAmount = swapTokensAtAmount;
}
```

Recommendation

Detected Arbitrary limits. If the threshold is set too low, it could result in frequent and unnecessary swaps, which would increase gas fees and potentially lead to losses due to slippage. On the other hand, if the threshold is set too high, it could result in liquidity being insufficient to handle large trades, which could negatively impact the token price and liquidity pool.

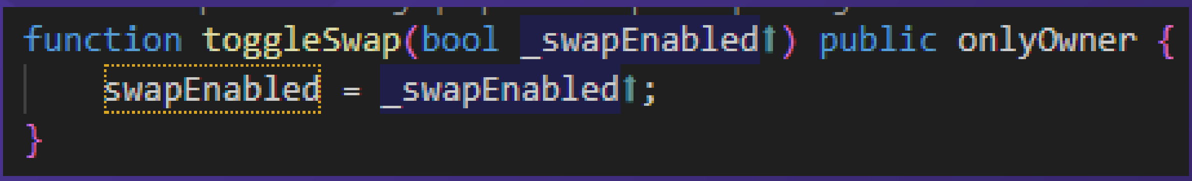
LOW RISK FINDING

Owner can change swap settings

Severity : Low

Overview

toggleSwap function allows the contract owner to enable or disable the automatic swapping of tokens for ETH.



```
function toggleSwap(bool _swapEnabled) public onlyOwner {
    swapEnabled = _swapEnabled;
}
```

Recommendation

It is recommended to ensure that the contract owner account is well secured and only accessible by authorized parties.

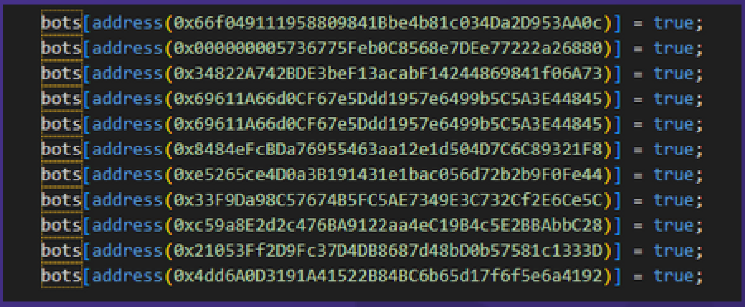
LOW RISK FINDING

Owner can't add an account to bot list after deploy but can add some address in the constructor.

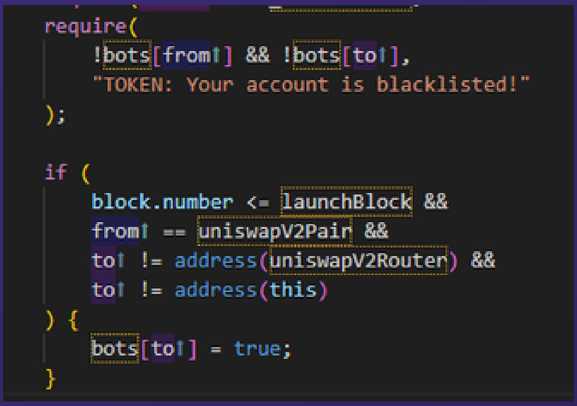
Severity : Low

Overview

The function then adds several addresses to the mapping by setting their boolean value to true. These addresses are likely addresses that are known to be associated with bots that are malicious or are being used for nefarious purposes. the function includes a check to prevent bot activity during the initial launch of the token. This is likely a measure to prevent bots from buying up the token during the initial launch and driving up the price artificially.



```
bots[address(0x66f049111958809841Bbe4b81c034Da2D953AA0c)] = true;
bots[address(0x000000005736775Feb0C8568e7DEe77222a26880)] = true;
bots[address(0x34822A742BDE3beF13acabF14244869841f06A73)] = true;
bots[address(0x69611A66d0CF67e5Ddd1957e6499b5C5A3E44845)] = true;
bots[address(0x69611A66d0CF67e5Ddd1957e6499b5C5A3E44845)] = true;
bots[address(0x8484eFcBDa76955463aa12e1d504D7C6C89321F8)] = true;
bots[address(0xe5265ce4D0a3B191431e1bac056d72b2b9F0Fe44)] = true;
bots[address(0x33F9Da98C57674B5FC5AE7349E3C732Cf2E6Ce5C)] = true;
bots[address(0xc59a8E2d2c476BA9122aa4eC19B4c5E2BBAbbC28)] = true;
bots[address(0x21053Ff2D9Fc37D4DB8687d48bD0b57581c1333D)] = true;
bots[address(0x4dd6A0D3191A41522B84BC6b65d17f6f5e6a4192)] = true;
```



```
require(
    !bots[from] && !bots[to],
    "TOKEN: Your account is blacklisted!"
);

if (
    block.number <= launchBlock &&
    from == uniswapV2Pair &&
    to != address(uniswapV2Router) &&
    to != address(this)
) {
    bots[to] = true;
}
```

Recommendation

–

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

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.

expelee

Building the Futuristic Blockchain Ecosystem