expelee

Building the Futuristic Blockchain Ecosystem

Security Audit Report

FOR

Gem AI

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	2 March 2023

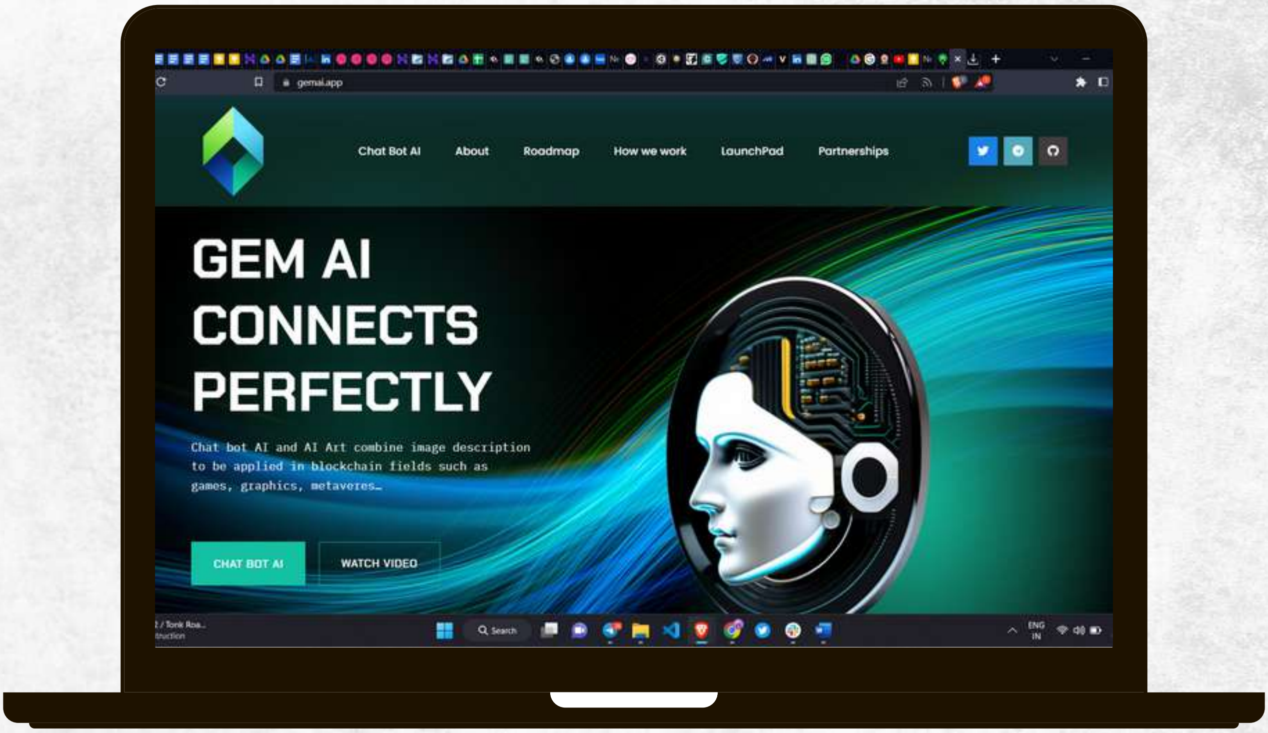
PROJECT DESCRIPTION

Gem AI

Why is it a gem ai the team develops advanced statistical and computational models such as machine learning, fuzzy logic and a knowledge base system that enables human-like computation, drawing, and perception; Different artificial intelligence techniques are used to perform different functions.

Social Media Profiles

Gem AI



GEM AI CONNECTS PERFECTLY

Chat bot AI and AI Art combine image description to be applied in blockchain fields such as games, graphics, metaveres_

https://gemai.app/

https://t.me/GemaiArb

https://twitter.com/Gemai_arb

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

-Team Expelee

CONTRACT DETAILS

Token Name

GEM AI

Symbol

GAI

Network

ARBITRUM

Language

Solidity

Contract Address (Verified)

0x2dC1320d7C11C0cF93D78A466bfA888Cab3076fd

Token Type

ERC20

Total Supply

1,000,000,000

Contract SHA-256 Checksum:

900adda7e7785dba36e8506d52cbc16c27be1460

Owner's Wallet

0x0FdC616949f2504501311A3ee28012992ea708D3

Deployer's Wallet

0x0FdC616949f2504501311A3ee28012992ea708D3

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

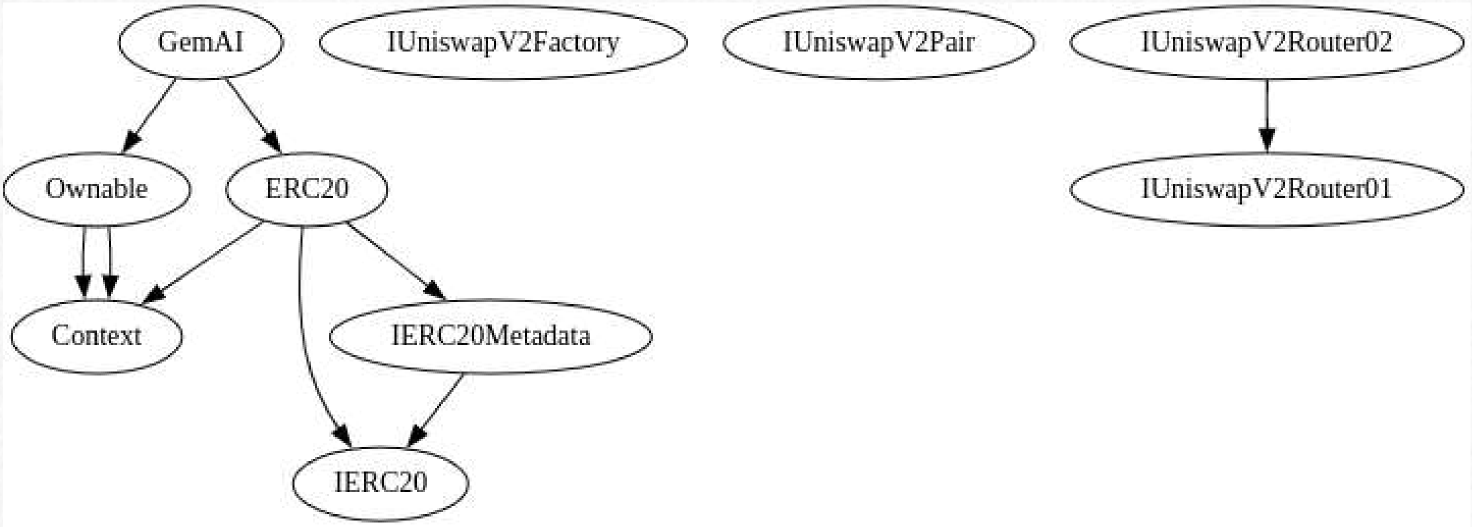
Used Tools:

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 able to set buy taxes up to 20%
- Owner is able to set sell taxes up to 30%
- Owner is able to set sell taxes up to 30%
- Owner is able to set max buy/sell amounts (min 0.1% of total supply)
- 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

1- Adding liquidity **(passed):**
https://testnet.bscscan.com/tx/0x1de1622d02169ce36822a326a9164b6f52c8b2cc0cebbc93b041834e44fcc123

2- Buying when excluded (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0x298a297990af4f6b38a15c87eb969bc67ef44e3712f9037aec17f391215583fc

3- Selling when excluded (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xc4d77b64dcd64b3c8138309c533e04c1c772c75ee1306ed74e737f2201d11956

4- Transferring when transfer without fee enabled (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xbea57fad55e0c1ecd617e102f56dced4e28d339eaa4c7d59b76543305c59858b

5- Buying when not excluded (2% tax) **(passed):**
https://testnet.bscscan.com/tx/0x3f2f5e32ab80c5e99f54b503ebdb38877f5f8542c9adbb73a1d43fcc470a142b

6- Selling when not excluded (2% tax) **(passed):**
https://testnet.bscscan.com/tx/0x50aca3751b82817e7acb6f03fed1d5f0cc8578afe9b7bea963acfd6e4897c7ef

Functional Tests

7- Transferring when transfer without fee disabled (2% tax) (passed):
https://testnet.bscscan.com/tx/0x3f7a28621ce15782a690b22ffb1147db0ed313d400972c4f67512ba17e50fa71

8- Internal swap (passed):
marketing wallet received ETH
https://testnet.bscscan.com/address/0xb3d8b55a92205591978e1c96f5eeb6dc3faed126#internaltx

MANUAL AUDIT

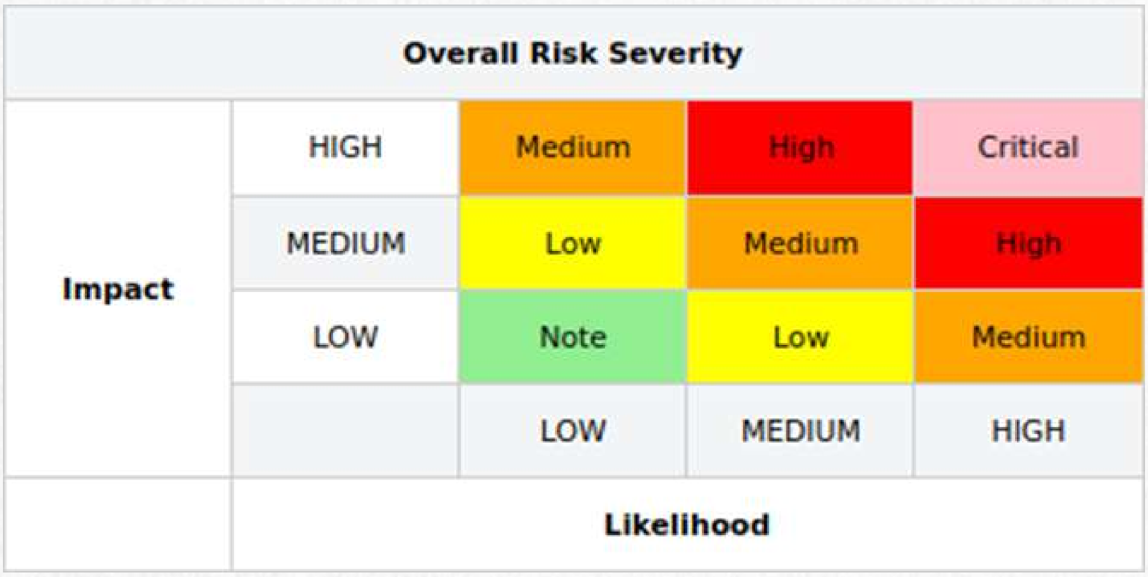
Severity Criteria

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			

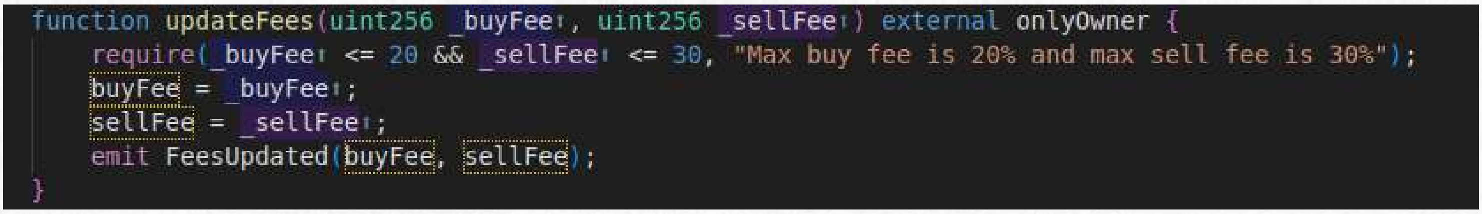
FINDINGS

- High Risk Findings:0
- Medium Risk Findings:0
- Low Risk Findings:2
- Suggestions & discussion: 0
- Gas Optimizations : 3

Low Risk Findings

Centralization

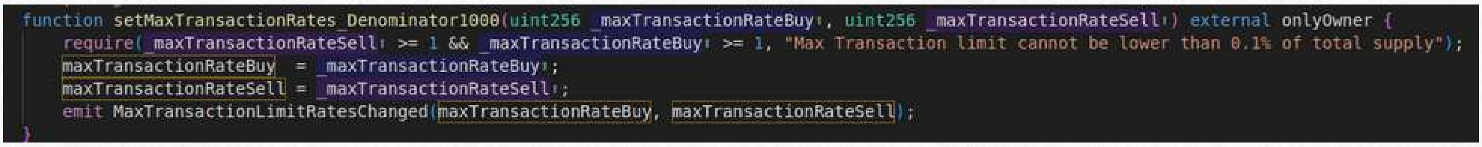
Owner has ability to change buy, sell fees:
Using updateFees function, owner is able to set buy taxes up to 20% and sell taxes up to 30%



```
function updateFees(uint256 _buyFee, uint256 _sellFee) external onlyOwner {
    require(_buyFee <= 20 && _sellFee <= 30, "Max buy fee is 20% and max sell fee is 30%");
    buyFee = _buyFee;
    sellFee = _sellFee;
    emit FeesUpdated(buyFee, sellFee);
}
```

Centralization

Owner is able to set a limit for max amount of buys and sells, this limit can not be less than 0.1% of total supply.



```
function setMaxTransactionRates_Denominator1000(uint256 _maxTransactionRateBuy, uint256 _maxTransactionRateSell) external onlyOwner {
    require(_maxTransactionRateSell >= 1 && _maxTransactionRateBuy >= 1, "Max Transaction limit cannot be lower than 0.1% of total supply");
    maxTransactionRateBuy  = _maxTransactionRateBuy;
    maxTransactionRateSell = _maxTransactionRateSell;
    emit MaxTransactionLimitRatesChanged(maxTransactionRateBuy, maxTransactionRateSell);
}
```

Gas Optimizations

- Use up to 3 indexed event arguments.
- define this variables as immutable:
  - uniswapV2Router
  - uniswapV2Pair
- define this variables as constant:
  - DEAD

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.