

expelee

Building the Futuristic **Blockchain Ecosystem**

Audit Report FOR



Doge Games

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	No KYC
 Audit Date	8 Sep 2022

Be aware that smart contracts deployed on the blockchain aren't resistant to internal exploit, external vulnerability, or hack. For a detailed understanding of risk severity, source code vulnerability, functional hack, and audit disclaimer, kindly refer to the audit.

- Team Expelee

PROJECT DESCRIPTION

Dogegames

Dogegames Is A Decentralized Mobile Gaming Protocol. Dogegames is proud to provide the first inclusive game field social platform, connecting gamers and game publishers worldwide at one platform where they can play, trade, & share all gaming items & experiences in a secure, transparent and accurate manner on the Dogechain Network.

 dogegames.co

 dogegamesco

 dogegamesco

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

- Team Expelee

CONTRACT DETAILS

Contract Name

Dogegames

Symbol

\$DGC

Contract Address

0x2A08296aBE6645b7dbAC8BF13a8fC37a0FD3428a

Network

**DOGECHAIN
(ERC-20 TOKEN)**

Language

Solidity

Total Supply

90,000,000,000 DGC

Decimals

18

Compiler

v0.8.7+commit.e28d00a7

License

Default license

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 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.

MANUAL AUDIT

- Owner is not able to set taxes more than 3% on buys / sells
- Owner is not able to disable trading
- Owner is able to **mint** new tokens

Centralization Issues

- **Critical**- owner is able to mint new tokens using mint function. owner is able to mint up to 600000000000 new tokens & send them to an arbitrary address(+66% new tokens)

```
function mint(address to, uint256 amount) public onlyOwner {
    uint256 realAmount = amount; // 10
    uint256 percentageToUse = getPercentageUsed(); // 0%
    uint256 amountToMint = realAmount - (( realAmount * percentageToUse )
/ 100 );
    if(super.totalSupply() < (150 * 10 ** 9 * (10 ** 18) )){
        if(super.totalSupply() + realAmount > (150 * 10 ** 9 * (10 **
18) )){
            amountToMint = ((150 * 10 ** 9) * (10 ** 18) ) -
super.totalSupply();
            hasReached = true;
        }

        if(!hasReached){
            _mint(to, amountToMint);
        }
    }
    else{
        hasReached = true;
    }
}
```

Recommendation:

give clear explanations about this mint function or remove it

Logical Issues

- **Critical**- in **transfer**, **transferFrom** and **mint** functions, there is no need to multiply **amount**(input) by 10^{18} , since it already has 18 decimals.

Recommendation:

this are bad practices, doing this reverts all transactions, delete 10^{**18}

```
uint256 realAmount = amount * 10 ** 18;  
_burn(creatorAddr, (amount * 10 ** 18 ));  
transactionFee = fee * 10 ** 18;  
uint256 transferAmount = amount * 10 ** 18;
```

Gas Optimizations

- **GO-1:** change this functions to external, **payOtherFee**, **payTransactionFee**, **setTransactionFeeAddress**, **setTranasactionFee**, **burn**, **mint**

Suggestions

- **S-1:** emit an event for this functions: **setTranasactionFee**, **setTransactionFeeAddress**, **mint**

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.