BAIL security

BAILSEC.IO

EMAIL : OFFICE@BAILSEC.IO

TWITTER : @BAILSECURITY

TELEGRAM : @HELLOATBAILSEC

FINAL REPORT:

DegenWin

June 2023

Disclaimer:

Security assessment projects are time-boxed and often reliant on information that may be provided by a client, its affiliates, or its partners. As a result, the findings documented in this report should not be considered a comprehensive list of security issues, flaws, or defects in the target system or codebase.

The content of this assessment is not an investment. The information provided in this report is for general informational purposes only and is not intended as investment, legal, financial, regulatory, or tax advice. The report is based on a limited review of the materials and documentation provided at the time of the audit, and the audit results may not be complete or identify all possible vulnerabilities or issues. The audit is provided on an "as-is," "where-is," and "as-available" basis, and the use of blockchain technology is subject to unknown risks and flaws.

The audit does not constitute an endorsement of any particular project or team, and we make no warranties, expressed or implied, regarding the accuracy, reliability, completeness, or availability of the report, its content, or any associated services or products. We disclaim all warranties, including the implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

We assume no responsibility for any product or service advertised or offered by a third party through the report, any open-source or third-party software, code, libraries, materials, or information linked to, called by, referenced by, or accessible through the report, its content, and the related services and products. We will not be liable for any loss or damages incurred as a result of the use or reliance on the audit report or the smart contract.

The contract owner is responsible for making their own decisions based on the audit report and should seek additional professional advice if needed. The audit firm or individual assumes no liability for any loss or damages incurred as a result of the use or reliance on the audit report or the smart contract. The contract owner agrees to indemnify and hold harmless the audit firm or individual from any and all claims, damages, expenses, or liabilities arising from the use or reliance on the audit report or the smart contract.

By engaging in a smart contract audit, the contract owner acknowledges and agrees to the terms of this disclaimer.

Executive Overview

This report been made for the DegenWin token. DegenWin is a premier online casino and sports betting platform which is fully licensed and based on the DGW Token

More information can be found here: https://degenwin.gitbook.io/degenwin/

1. Project Details

Project	Cairo Finance
Website	https://degenwin.com/
Type	ERC20
Language	Solidity
Methods	Manual Review
Github	1. https://github.com/DegenWin/Token/blob/main/DGW%20Token%20Pre%20Audit 2. https://github.com/DegenWin/Token/blob/main/DGW%20Token%20Final

2. Detections Overview

Severity	Found	Resolved	Partially Resolved	Acknowledged (no change made)
High	0			
Medium	0			
Low	0			
Informational	3	3		
Total	3	3		

2.1 Detections Definitions

Severity	Description
High	The problem poses a significant threat to the confidentiality of a considerable number of users' sensitive data. It also has the potential to cause severe damage to the client's reputation or result in substantial financial losses for both the client and the affected users.
Medium	While medium level vulnerabilities may not be easy to exploit, they can still have a major impact on the execution of a smart contract. For instance, they may allow public access to critical functions, which could lead to serious consequences.
Low	Poses a very low level risk to the project or users. Nevertheless the issue should be fixed immediately
Informational	Effects are small and do not post an immediate danger to the project or users

3. Detections

The **DegenWin** token is a simple ERC20 https://github.com/OpenZeppelin/openzeppelincontracts/blob/master/contracts/token/ERC20/ERC20.sol) token with a max supply of 11400000000 tokens. All tokens are pre-minted during contract deployment to the **_mintAddress**. This address will custody all tokens until further notice and it is not possible to mint additional tokens.

The contract is extended by the **ERC20Burnable** extension (https://github.com/OpenZeppelin/openzeppelincontracts/blob/master/contracts/token/ERC20/extensions/ERC20Burnable.sol) which allows users to burn their own tokens and approved addresses to burn tokens from the corresponding address.

Furthermore, the contract inherits OpenZeppelin's **Ownable** library, however, it is not actively used.

Issue	Ownable library is unused
Severity	**Informational**
Description	The above mentioned library is inherited by the DegenWin contract but is not actively used.
Recommendations	Think about the use case of this function, if it in fact serves a purpose and consider implementing this purpose. If it does not serve any purpose it should be removed.
Comments	The library has been removed.

Issue	_beforeTokenTransfer function is unnecessary
Severity	**Informational**
Description	The above mentioned function does not serve any use case, it is simply calling the function within the ERC20 contract without any additional checks or calls.
Recommendations	Think about the use case of this function, if it in fact serves a purpose and consider implementing this purpose. If it does not serve any purpose it should be removed.
Comments	The function has been removed.

Issue	ERC20 inheritance is unnecessary
Severity	**Informational**
Description	Since the **ERC20Burnable** extension already inherits the **ERC20** contract, it is not necessary to inherit **ERC20** twice. *abstract contract ERC20Burnable is Context, ERC20*
Recommendations	Remove the ERC20 inheritance.
Comments	The inheritance has been removed.