CoinDogg

Audit Report

May 2021

Table of Contents

Overview

CoinDogg is an ERC20 compatible token with token transfer, hold token, check balance functionality. This contract follows all ERC20 Standard protocols. Smart contract checks all the balances, addresses before performing any operation.

This smart contract is designed with ownable functions to manage the owner of the smart contract.

Tokens are compatible with all types of ETH supporting wallets. These tokens can be used on all standard trading, staking or exchange platforms.

Project Engagement

During May of 2021, **CoinDogg** engaged Solidproof.io to audit smart contracts that they created. The engagement was technical in nature and focused on identifying security flaws in the design and implementation of the contracts. **CoinDogg** provided Solidproof.io with access to their code repository and whitepaper.

Description

The CoinDogg token will act as one of the means of value exchange for fans to support their favorite creators. Additionally, it will be used to trade NFTs on CoinDogg's marketplace and by creators to commission freelance artists to create artwork that will be minted to NFTs for fans.

Logo

Contract Link

https://etherscan.io/address/0x620389053d327a103b573b414be1aab0fbed32b8#code

Audit Details and Target

Target of the Audit

The scope of this audit was to analyse **CoinDogg** smart contracts codebase for quality, security, and correctness.

Checked Vulnerabilities

The smart contract is scanned and checked for multiple types of possible bugs and issues. This mainly focuses on issues regarding security, attacks, mathematical errors, logical and business logic issues.

The results are shown on the results page.

Techniques and Methods

Structual Analysis
In this step we have analyzed the design patterns and structure of smart contracts. A thorough check was done to ensure the smart contract is structured in a way that will not result in future problems.

Static Analysis
Static Analysis of Smart Contracts was done to identify contract vulnerabilities. In this step a series of automated tools are used to test security of smart contracts.

Manual Analysis
Manual Analysis or review of code was done to identify new vulnerability or verify the vulnerabilities found during the static analysis. Contracts were completely manually analyzed, their logic was checked and compared with the one described in the whitepaper. Besides, the results of automated.

Gas Consumption
In this step we have checked the behaviour of smart contracts in production. Checks were done to know how much gas gets consumed and possibilities of optimization of code to reduce gas consumption.

Tools and Platforms used for Audit
Remix IDE, Truffle, Mythril, Slither, SmartCheck.

Audit Results

<div style="background-color:#c6e0b4; text-align:center; padding:40px;">

Audit passed

</div>

Issue Categories

High severity Issue
In this step we have analyzed the design patterns and structure of smart contracts. A thorough check was done to ensure the smart contract is structured in a way that will not result in future problems.

Medium level severity Issue
Static Analysis of Smart Contracts was done to identify contract vulnerabilities. In this step a series of automated tools are used to test security of smart contracts.

Low level severity Issue
Manual Analysis or review of code was done to identify new vulnerability or verify the vulnerabilities found during the static analysis. Contracts were completely manually analyzed, their logic was checked and compared with the one described in the whitepaper. Besides, the results of automated.

Issues found

	High level	Medium level	Low level
Open	0	0	0
Closed	0	0	0

Detailed Information for may found issues are in the Audit Protocol

Informational

- No informational comments -

Summary

The contract does not contain issues of high or medium criticality and it's safe to deploy.

Attachments

1. Passed Audit Certificate

2. Audit Protocol

Disclaimer

Solidproof.io

Trust Made in Germany