

# DedaCoin



**BSC**



**Ethereum**



**Tron**

## Public audit report



[trustedbytes.org](http://trustedbytes.org)



[contact@trustedbytes.org](mailto:contact@trustedbytes.org)



# Contents

<b>1. Project description.....</b>	<b>3</b>
<b>2. Introduction.....</b>	<b>4</b>
<b>3. Technical audit.....</b>	<b>5</b>
<b>3.1 Automatic check results.....</b>	<b>5</b>
<b>3.2 Issues.....</b>	<b>6</b>
<b>4. Project Overview.....</b>	<b>7</b>
<b>5. Disclaimer.....</b>	<b>8</b>



# Project description

<b>Token name</b>	<b>DEDA</b>
<b>URL</b>	<b><a href="https://dedacoin.co/">https://dedacoin.co/</a></b>
<b>Blockchain</b>	<b>BSC, ETH, TRON</b>
<b>Project name</b>	<b>DedaCoin</b>
<b>Logo</b>	
<b>Address erc20</b>	<b>0x15F9EB4b9BEaFa9Db35341c5694c0b6573809808</b>
<b>Address TRON</b>	<b>TZ6gek9k3F4jke9j3DVeiycEXy8NJKTxQc</b>
<b>Date</b>	<b>2024-09-07</b>



# Introduction

**The report was prepared for DedaCoin team. The audit was requested by a community member.**

**The token supports the BSC, ERC, TRON interface. The token contract is inherited from OpenZeppelin's ERC20 implementation.**

## Procedure

**We perform our audit according to the following procedure:**

- Automated analysis;
- Scanning the project's smart contracts with automated Solidity analysis tools of our own development;
- Manual verification all the issues found by the tools.

## Manual audit

- ◊ Manually analyse smart contracts for well known or trended security vulnerabilities
- ◊ Smart contracts' logic check



# Automatic check results

Title	Check result
Unencrypted Private Data On-Chain	passed
Code With No Effects	passed
Message call with hardcoded gas amount	passed
Typographical Error	passed
DoS With Block Gas Limit	passed
Presence of unused variables	passed
Incorrect Inheritance Order	passed
Requirement Violation	passed
Weak Sources of Randomness from Chain	passed
Attributes	passed
Shadowing State Variables	passed
Incorrect Constructor Name	passed
Block values as a proxy for time	passed
Authorization through tx.origin	passed
DoS with Failed Call	passed
Delegatecall to Untrusted Callee	passed
Use of Deprecated Solidity Functions	passed
Assert Violation	passed
State Variable Default Visibility	passed
Reentrancy	passed
Unprotected SELFDESTRUCT Instruction	passed

 **Issues**

High risk	Not found
Medium risk	Not found
Low risk	Not found

**Security Score:** A

## Issues description:

Issues not found



## Project overview



Twitter

**21,8k  
subscribers**



Coinmarketcap

**rating: #3004**



Telegram

**33k  
members**



Discord

**1,5k  
members**



# Disclaimer

This audit report is based on the existing knowledge and skills of our auditors, as well as specialized tools of our own development. Verification is always carried out for already known main types of vulnerabilities and threats, both from the user (investor) and from the developers. An audit of a smart contract doesn't provide a full guarantee that a particular project is completely safe. An audit is just one of the tools to check the honesty of the founders (developers) and their professional skills in developing, along with such methods as liquidity pool locking , KYC and others. Only in the aggregate, you can decide whether to invest in a project or not to do it. First of all DYOR and nothing else.



**Trustedbytes.org**



**Contact@trustedbytes.org**



**/Trustedbytesorg**



**@trustedbytes**



**@trustedbytesorg**