Audit Report for **AntiBotStandardToken**

Date: 20 November 2024

Audit result: **Passed**

**Token Address:** 0x518Eb73A88060972b19d909d515054DBd9b02E05

**Name:** DealGuard

**Symbol:** DGT

**Decimals**: 18

**Network:** Polygon Scan

**Token Type**: ERC-20

**Owner**: 0x0276d3AE91F43B3DaA6d6E2A0802AC2622bCC82B

**Deployer:** 0x0276d3AE91F43B3DaA6d6E2A0802AC2622bCC82B

**Token Supply:** 210,000,000

**Checksum:** abbca886282c61687cdccb75bb704474

**Token Overview:**

**Buy Fee:** 0-0%

**Sell Fee:** 0-0%

**Transfer Fee:** 0-0%

**Fee Privilege:** No

**Ownership:** Owned

**Minting:** No

**Max Tx:** No  
**Blacklist:** No

**Static Analysis**

A static analysis of the code was performed using Slither. No issues were found.





**Ownership Privileges:**

- The owner can set enable anti-bot.

**Findings:**

**Critical**: 0

**High**: 0

**Medium**: 0

**Low**: 2

**Informational** **&** **Optimizations**: 0

**Centralization – Missing Zero Address**

**Severity**: **Low**

**Status:** Open

**Overview:**

functions can take a zero address as a parameter (0x00000...). If a function parameter of address type is not properly validated by checking for zero addresses, there could be serious consequences for the contract's functionality.

**constructor(**

**string memory name\_,**

**string memory symbol\_,**

**uint8 decimals\_,**

**uint256 totalSupply\_,**

**address pinkAntiBot\_,**

**address serviceFeeReceiver\_,**

**uint256 serviceFee\_**

**) payable {**

**\_name = name\_;**

**\_symbol = symbol\_;**

**\_decimals = decimals\_;**

**\_mint(owner(), totalSupply\_);**

**pinkAntiBot = IPinkAntiBot(pinkAntiBot\_);**

**pinkAntiBot.setTokenOwner(owner());**

**enableAntiBot = true;**

**emit TokenCreated(**

**owner(),**

**address(this),**

**TokenType.antiBotStandard,**

**VERSION**

**);**

**payable(serviceFeeReceiver\_).transfer(serviceFee\_);**

**}**

**Suggestion:**

It is suggested that the address should not be zero or dead.

**Centralization – Remove the safe math library.**

**Severity**: **Low**

**Status:** Open

**Line Number: 205-416**

**Overview:**

The Safe Math library is no longer needed for Solidity version 0.8 and above. This is because Solidity 0.8 includes checked arithmetic operations by default. All of Safe Math’s methods are now inherited into Solidity programming.