SLOWMIST

Smart Contract Security Audit Report

The SlowMist Security Team received the team's application for smart contract security audit of the Chainbase Token on 2024.11.11. The following are the details and results of this smart contract security audit:

Token Name :

Chainbase Token

The contract address :

https://github.com/chainbase-labs/chainbase-contract/blob/main/src/ChainbaseToken.sol

commit: c8a35c16813f81d9e06858f29e9bbbadfe5c4c05

The audit items and results :

(Other unknown security vulnerabilities are not included in the audit responsibility scope)

NO.	Audit Items	Result
1	Replay Vulnerability	Passed
2	Denial of Service Vulnerability	Passed
3	Race Conditions Vulnerability	Passed
4	Authority Control Vulnerability Audit	Passed
5	Integer Overflow and Underflow Vulnerability	Passed
6	Gas Optimization Audit	Passed
7	Design Logic Audit	Passed
8	Uninitialized Storage Pointers Vulnerability	Passed
9	Arithmetic Accuracy Deviation Vulnerability	Passed
10	"False top-up" Vulnerability	Passed
11	Malicious Event Log Audit	Passed
12	Scoping and Declarations Audit	Passed
13	Safety Design Audit	Passed

NO.	Audit Items	Result
14	Non-privacy/Non-dark Coin Audit	Passed

Audit Result : Passed

Audit Number : 0X002411120001

Audit Date : 2024.11.11 - 2024.11.12

Audit Team : SlowMist Security Team

Summary conclusion : This is a token contract that does not contain the token vault section and the dark coin functions. The total amount of contract tokens can be changed, users can call the burn and burnFrom functions to burn their tokens. The contract does not have the Overflow and the Race Conditions issue.

During the audit, we found the following information:

1. The owner role can mint tokens through the mint function and there is an upper limit on the amount of tokens.

The source code:

```
// SPDX-License-Identifier: MIT
//SlowMist// The contract does not have the Overflow and the Race Conditions issue
pragma solidity ^0.8.22;

import "@openzeppelin/contracts/access/Ownable.sol";
import "@openzeppelin/contracts/token/ERC20/extensions/ERC20Capped.sol";
import "@openzeppelin/contracts/token/ERC20/extensions/ERC20Burnable.sol";

contract ChainbaseToken is ERC20Capped, ERC20Burnable, Ownable {
    constructor() ERC20("Chainbase Token", "C") ERC20Capped(2100 * 10 ** 26) {}
    //SlowMist// The owner role can mint tokens through the mint function and there is
an upper limit on the amount of tokens
    function mint(address to, uint256 amount) public onlyOwner {
        _mint(to, amount);
    }

    function _mint(address account, uint256 amount) internal override(ERC20Capped,
ERC20) {
        ERC20Capped._mint(account, amount);
    }
}
```

Statement

SlowMist issues this report with reference to the facts that have occurred or existed before the issuance of this report, and only assumes corresponding responsibility based on these.

For the facts that occurred or existed after the issuance, SlowMist is not able to judge the security status of this project, and is not responsible for them. The security audit analysis and other contents of this report are based on the documents and materials provided to SlowMist by the information provider till the date of the insurance report (referred to as "provided information"). SlowMist assumes: The information provided is not missing, tampered with, deleted or concealed. If the information provided is missing, tampered with, deleted, concealed, or inconsistent with the actual situation, the SlowMist shall not be liable for any loss or adverse effect resulting therefrom. SlowMist only conducts the agreed security audit on the security situation of the project and issues this report. SlowMist is not responsible for the background and other conditions of the project.

SLOWMIST

Official Website

www.slowmist.com

E-mail

team@slowmist.com

Twitter

@SlowMist_Team

Github

https://github.com/slowmist