verichains

SECURITY AUDIT OF

CREDITDEFI TOKEN SMART CONTRACT

CREDIT DEFI

Public Report

Nov 01, 2022

Verichains Lab

info@verichains.io

https://www.verichains.io

Driving Technology > Forward

ABBREVIATIONS

Name	Description
Ethereum	An open source platform based on blockchain technology to create and distribute smart contracts and decentralized applications.
Ether (ETH)	A cryptocurrency whose blockchain is generated by the Ethereum platform. Ether is used for payment of transactions and computing services in the Ethereum network.
Smart contract	A computer protocol intended to digitally facilitate, verify or enforce the negotiation or performance of a contract.
Solidity	A contract-oriented, high-level language for implementing smart contracts for the Ethereum platform.
Solc	A compiler for Solidity.
ERC20	ERC20 (BEP20 in Binance Smart Chain or *x*RP20 in other chains) tokens are blockchain-based assets that have value and can be sent and received. The primary difference with the primary coin is that instead of running on their own blockchain, ERC20 tokens are issued on a network that supports smart contracts such as Ethereum or Binance Smart Chain.

EXECUTIVE SUMMARY

This Security Audit Report was prepared by Verichains Lab on Nov 01, 2022. We would like to thank the CreditDefi for trusting Verichains Lab in auditing smart contracts. Delivering high-quality audits is always our top priority.

This audit focused on identifying security flaws in code and the design of the CreditDefi Token Smart Contract. The scope of the audit is limited to the source code files provided to Verichains. Verichains Lab completed the assessment using manual, static, and dynamic analysis techniques.

During the audit process, the audit team had identified no vulnerable issue in the contract code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About CreditDefi Token Smart Contract

CreditDefi is a decentralized exchange platform built on top of CDF a new type of platform that allows CreditDefi to be more security and to avoid attacking from hackers in transactions and storage electronic money. CreditDefi position as an exchange that can build and develop a decentralized database, realizing the essence of a decentralized trading system network, as the essence of DEFI.

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of the smart contracts of CreditDefi Token Smart Contract. It was conducted on the source code provided by the CreditDefi team.

It was conducted on commit `7a3b9710411879770478f125a8d488be6c286484` from git repository *https://github.com/CreditDefi2022/smartcontract-CDF/*.

The file in our audit scope is `CDF.sol`

1.3. Audit methodology

Our security audit process for smart contract includes two steps:

- Smart contract codes are scanned/tested for commonly known and more specific vulnerabilities using public and RK87, our in-house smart contract security analysis tool.
- Manual audit of the codes for security issues. The contracts are manually analyzed to look for any potential problems.

Following is the list of commonly known vulnerabilities that were considered during the audit of the smart contract:

- Integer Overflow and Underflow
- Timestamp Dependence
- Race Conditions
- Transaction-Ordering Dependence
- DoS with (Unexpected) revert
- DoS with Block Gas Limit
- Gas Usage, Gas Limit and Loops
- Redundant fallback function
- Unsafe type Inference
- Reentrancy

- Explicit visibility of functions state variables (external, internal, private and public)
- Logic Flaws

For vulnerabilities, we categorize the findings into categories as listed in table below, depending on their severity level:

SEVERITY LEVEL	DESCRIPTION
CRITICAL	A vulnerability that can disrupt the contract functioning; creates a critical risk to the contract; required to be fixed immediately.
HIGH	A vulnerability that could affect the desired outcome of executing the contract with high impact; needs to be fixed with high priority.
MEDIUM	A vulnerability that could affect the desired outcome of executing the contract with medium impact in a specific scenario; needs to be fixed.
LOW	An issue that does not have a significant impact, can be considered as less important.

Table 1. Severity levels

1.4. Disclaimer

Please note that security auditing cannot uncover all existing vulnerabilities, and even an audit in which no vulnerabilities are found is not a guarantee for a 100% secure smart contract. However, auditing allows discovering vulnerabilities that were unobserved, overlooked during development and areas where additional security measures are necessary.

2. AUDIT RESULT

2.1. Overview

The CreditDefi Token Smart Contract was written in `Solidity` language, with the required version to be `^0.8.0`. The source code was written based on OpenZeppelin's library.

2.1.1. Token contract

This is the ERC20 token contract in the CreditDefi Token Smart Contract, which extends `Context`, `ERC20` and `Ownable` contracts. With `Ownable`, by default, Contract Owner is the contract deployer, but he can transfer ownership to another address at any time. The contract pre-minted all 10 million tokens to the Contract Owner when deployed.

The contract supports taking fee while transferring tokens (buy, sell and transfer) which can customized by the Contract Owner at any time. The contract also supports blacklisting addresses. The Contract Owner can add/remove addresses to/from blacklist and blacklisted addresses can't transfer their tokens.

Table below lists some properties of the audited CreditDefi token contract (as of the report writing time).

PROPERTY	VALUE
Name	CreditDefi
Symbol	CDF
Decimals	18
Total Supply	10,000,000,000 ($x10^{18}$) Note: the number of decimals is 18, so the total representation token will be 10,000,000,000 or 10 billion.

Table 2. CreditDefi Token properties

2.2. Findings

During the audit process, the audit team found no vulnerability in the given version of CreditDefi Token Smart Contract.

APPENDIX

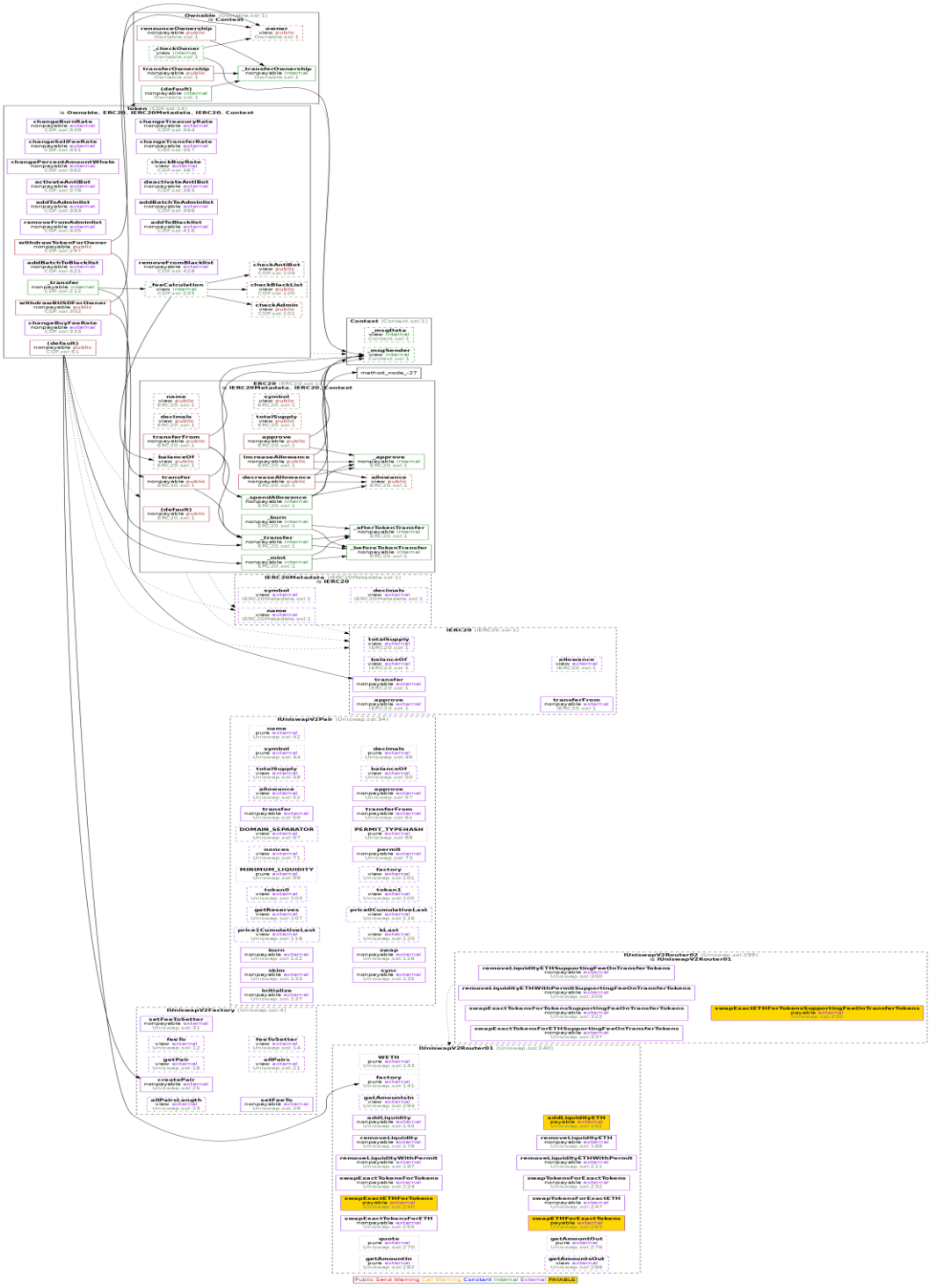

Image 1. CreditDefi Token Smart Contract call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Nov 01, 2022*	Public Report	Verichains Lab

Table 3. Report versions history