SeaRoad Token Smart Contract Audit

1. Introduction

iosiro was commissioned by CROAD International to conduct a smart contract audit on their SeaRoad ERC-20 token. The audit was performed on 05 December 2020.

This report is organized into the following sections.

- **Section 2 - Executive Summary:** A high-level description of the findings of the audit.
- **Section 3 - Audit Details:** A description of the scope and methodology of the audit.
- **Section 4 - Design Specification:** An outline of the intended functionality of the smart contracts.
- **Section 5 - Detailed Findings:** Detailed descriptions of the findings of the audit.

The information in this report should be used to better understand the risk exposure of the smart contracts, and as a guide to improving the security posture of the smart contracts by remediating issues identified. The results of this audit are only a reflection of the source code reviewed at the time of the audit and of the source code that was determined to be in-scope.

The purpose of this audit was to achieve the following:

- Identify potential security flaws.
- Ensure that the smart contracts adhered to the ERC-20 standard.

Assessing the off-chain functionality associated with the contracts, for example, backend web application code, was out of scope of this audit.

Due to the unregulated nature and ease of transfer of cryptocurrencies, operations that store or interact with these assets are considered very high risk with regards to cyber attacks. As such, the highest level of security should be observed when interacting with these assets. This requires a forward-thinking approach, which takes into account the new and experimental nature of blockchain technologies. Strategies that should be used to encourage secure code development include:

- Security should be integrated into the development lifecycle and the level of perceived security should not be limited to a single code audit.
- Defensive programming should be employed to account for unforeseen circumstances.
- Current best practices should be followed where possible.

2. Executive Summary

This report presents the findings of an audit performed by iosiro on the ERC-20 Sea Road Token.

No issues were identified during the audit. The code was almost entirely based on the OpenZeppelin v2.3.0 ERC-20 token implementation, barring a few minor changes. As such, the code conformed to general best practices and was of a high standard.

3. Audit Details

3.1 Scope

The source code considered in-scope for the assessment is described below. Code from all other files is considered to be out-of-scope. For the purpose of this audit, out-of-scope code that interacts with in-scope code is assumed to function as intended and introduce no functional or security vulnerabilities.

3.1.1 Smart Contracts

- **Contract Name:** TokenMintERC20Token
- **Address:** 0x4202f32b18742acf0a566ea78726eac3bc93ec70

3.2 Methodology

A variety of techniques, described below, were used to conduct the audit.

3.2.1 Code Review

The source code was manually inspected to identify potential security flaws. Code review is a useful approach for detecting security flaws, discrepancies between the specification and implementation, design improvements, and high risk areas of the system.

3.2.2 Dynamic Analysis

The contracts were compiled, deployed, and manually tested in a Ganache test environment. Manual analysis was used to confirm that the code operated at a functional level and to verify the exploitability of any potential security issues identified.

3.2.3 Automated Analysis

Tools were used to automatically detect the presence of several types of security vulnerabilities, including reentrancy, timestamp dependency bugs, and transactionordering dependency bugs. The static analysis results were manually analyzed to remove false-positive results. True positive results would be indicated in this report.

Static analysis tools commonly used include Slither, Securify, and MythX. Tools such as the Remix IDE, compilation output, and linters could also be used to identify potential areas of concern.

3.3 Risk Ratings

Each issue identified during the audit has been assigned a risk rating. The rating is determined based on the criteria outlined below.

- **High Risk** - The issue could result in a loss of funds for the contract owner or system users.
- **Medium Risk** - The issue resulted in the code specification being implemented incorrectly.
- **Low Risk** - A best practice or design issue that could affect the security of the contract.
- **Informational** - A lapse in best practice or a suboptimal design pattern that has a minimal risk of affecting the security of the contract.
- **Closed** - The issue was identified during the audit and has since been addressed to a satisfactory level to remove the risk that it posed.

4. Design Specification

The following section outlines the intended functionality of the system at a high level. The specification is based on the implementation in the codebase and any perceived points of conflict should be highlighted with the auditing team to determine the source of the discrepancy.

Overview

Sea Road Token is an ERC-20 token with the following parameters.

Field	Value
Symbol	SRT
Name	Sea Road Token
Decimals	18
Initial Supply	1 billion

Additional SRT tokens cannot be minted. The initial supply was wholly minted to the contract creator's address.

Source Code

The Sea Road Token smart contract has the following inheritance structure.

```
└── TokenMintERC20Token
    └── ERC20
        └── IERC20
```

5. Detailed Findings

The following section details the findings of the audit.

5.1 High Risk

No high risk issues were present at the conclusion of the review.

5.2 Medium Risk

No medium risk issues were present at the conclusion of the review.

5.3 Low Risk

No low risk issues were present at the conclusion of the review.

5.4 Informational

No informational issues were present at the conclusion of the review.

5.5 Closed

No issues were closed during the audit.

AUDIT REPORTS