Tokenine co., ltd.
39/129 moo 7, Suan Koon Village
Ban Suan, Muang Chonburi, Chonburi, 20000

VeloLabs Token Controller

Smart Contract Audit Report

Prepared By: Sutee Sudprasert

Tokenine, Thailand

May 24th, 2023

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to: (i) cybersecurity vulnerabilities and issues in the smart contract source code analyzed, the details of which are set out in this report, (Source Code); and (ii) the Source Code compiling, deploying and performing the intended functions. In order to get a full view of our findings and the scope of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Tokenine and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Tokenine) owe no duty of care towards you or any other person, nor does Iosiro make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Tokenine hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Tokenine hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Tokenine, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

Document Properties

Client	VeloLabs
Title	Smart Contract Audit Report
Repository	https://gitlab.com/velolabs/smart_contract/mint_token_controller/
Commit	eb14e6bf2499f0240ef464654fca60a28767e167
Author	Sutee Sudprasert
Auditors	Sutee Sudprasert
Reviewed By	Thanarat Kuawattanaphan
Approved By	Dom Charoenyos
Classification	Confidential

Introduction

Thai Chain was contracted by VeloLab to conduct an audit of smart contracts. The report presents the findings of the security assessment of the smart contracts and its code review conducted at May 24th, 2023

Scope

The scope of the project is smart contracts in the repository:

https://gitlab.com/velolabs/smart_contract/mint_token_controller

Executive Summary

VeloLabs Token Controller is a multi-chain token controller. It was designed for deploying a new token and minting the token and the transactions are signed by using EIP-712 standard. The deployed tokens can be transferred between the controllers.

The project consists of 2 main contracts that are Token.sol and TokenCtrl.sol. Token.sol is basically an ERC-20 that was modified by adding the “initialize function” to support the Clone pattern. TokenCtrol.sol is the controller that contains all of the business logic of the project.

Our team performed static analysis, code functionality and manual audit. We found 3 issues during the audit.

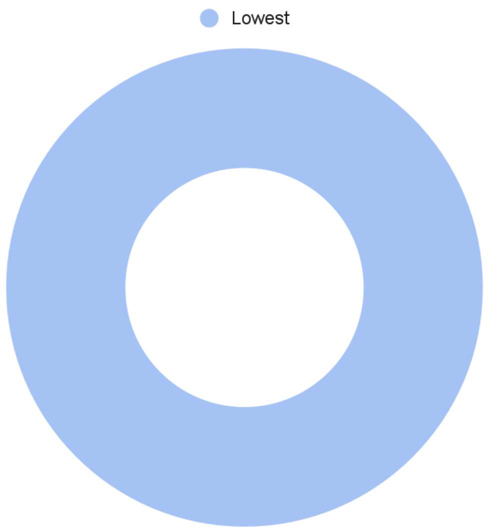
Severity Definitions

Severity Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to asset loss or data manipulations.
High	High-level vulnerabilities have a significant impact on smart contract execution, e.g., public access to crucial functions.
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to asset loss or data manipulations.
Low	Low-level vulnerabilities are mostly related to outdated, unused, etc. code snippets that can't have a significant impact on execution.
Lowest / Coding Style / Best Practice	Lowest-level vulnerabilities, code style violations, and info statements can't affect smart contract execution and can be ignored.

Findings

3 Total Issues

Critical	0 (0%)
High	0 (0%)
Medium	0 (0%)
Low	0 (0%)
Lowest	3 (100%)



ID	Title	Category	Severity	Status
VELO-01	Inconsistency of boolean expressions	Best Practice	Lowest	Acknowledged
VELO-02	Incorrect code comments	Coding Style	Lowest	Acknowledged
VELO-03	Misleading function name	Coding Style	Lowest	Acknowledged

Audit Overview

Critical

No critical issues were found.

High

No critical issues were found.

Medium

No critical issues were found.

Low

No low issues were found.

Lowest / Coding Style / Best Practice

1. **VELO-01**: Line 329 and 422 of TokenCtrl.sol. Those two boolean expressions have the same condition but were written in different styles.

   Line 329: `authorities[newOwner] == false`
   Line 422: `!authorities[authority_]`

   Recommendation
   We recommend using only one coding style to improve the code readability.

2. **VELO-02**: The following functions use `onlyAuthority` modifier but “caller must be **owner**” are written in the code comments.
   - `changeTokenAuthority`
   - `enlistToken`
   - `delistToken`

   Recommendation
   The code comments should be changed to “caller must be **authority**”.

3. **VELO-03**: The function `changeTokenAuthority` is for changing the owner of the deployed token. Basically the owner is referred to the controller itself. But “authority” wording which is widely used in the code represents the given wallet address.

   Recommendation
   We suggest to rename the function to `changeTokenController` and also refactor all of the related statements such as the second parameter should be named to `newController_`, the event `ChangeTokenAuthority` should be named to `ChangeTokenController`, and so on.