TomoChain

SECURITY AUDIT OF

MOLLECTOR TOKEN SMART CONTRACT

Public Report

Mar 28, 2022

TomoChainLab

admin@tomochain.com

https://tomochain.com/

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 prepared by TomoChainLab on March 28, 2022. We would liketo thank the Mollector for trusting TomoChain 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 Mollector Token Smart Contract. The scope of the audit is limited to the source code files provided to TomoChain. TomoChainLab completed the assessment using manual, static, and dynamic analysis techniques.

During the audit process, the audit team had identified no vulnerable issues in the smart contracts code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About Mollector Token Smart Contract

Mollector is a mystical-themed and tactical NFT card game. Mollector takes place in a fantasy universe full of wonder and also fraught with chaos, hosting a seemingly infinite amount of worlds all striving for supremacy and sometimes, for mere existence.

Molecule is an ERC20 token that Mollector players can use in the game. By owning Molecule, users can use it as the fusion fee, receiving additional rewards upon staking $MOL, accelerator for generating Mutant Essence and Purchasing chests and cards on the in-game marketplace

1.2. Audit scope

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

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 was 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

Table 2 lists some properties of the audited Mollector Token Smart Contract (as of the report writing time).

PROPERTY	VALUE
Name	Molecule
Symbol	MOL
Decimals	18
Total Supply	1,000,000,000 (x10^{18}) Note: the number of decimals is 18, so the total representation token will be 1,000,000,000 or 1 billion.

Table 2. The Mollector Token Smart Contract properties

2.2. Contract codes

The Mollector Token Smart Contract was written in Solidity language, with the required version to be 0.8.10.

The contract extends ERC20, Ownable and ERC20Pausable contracts. With Ownable, by default, Token Owner is contract deployer but he can transfer ownership to another address at any time. ERC20Burnable allows token holders to destroy their own tokens. Token Owner can pause/unpause contract using ERC20Pausable contract, user can only transfer tokens when contract is not paused.

2.3. Findings

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

APPENDIX

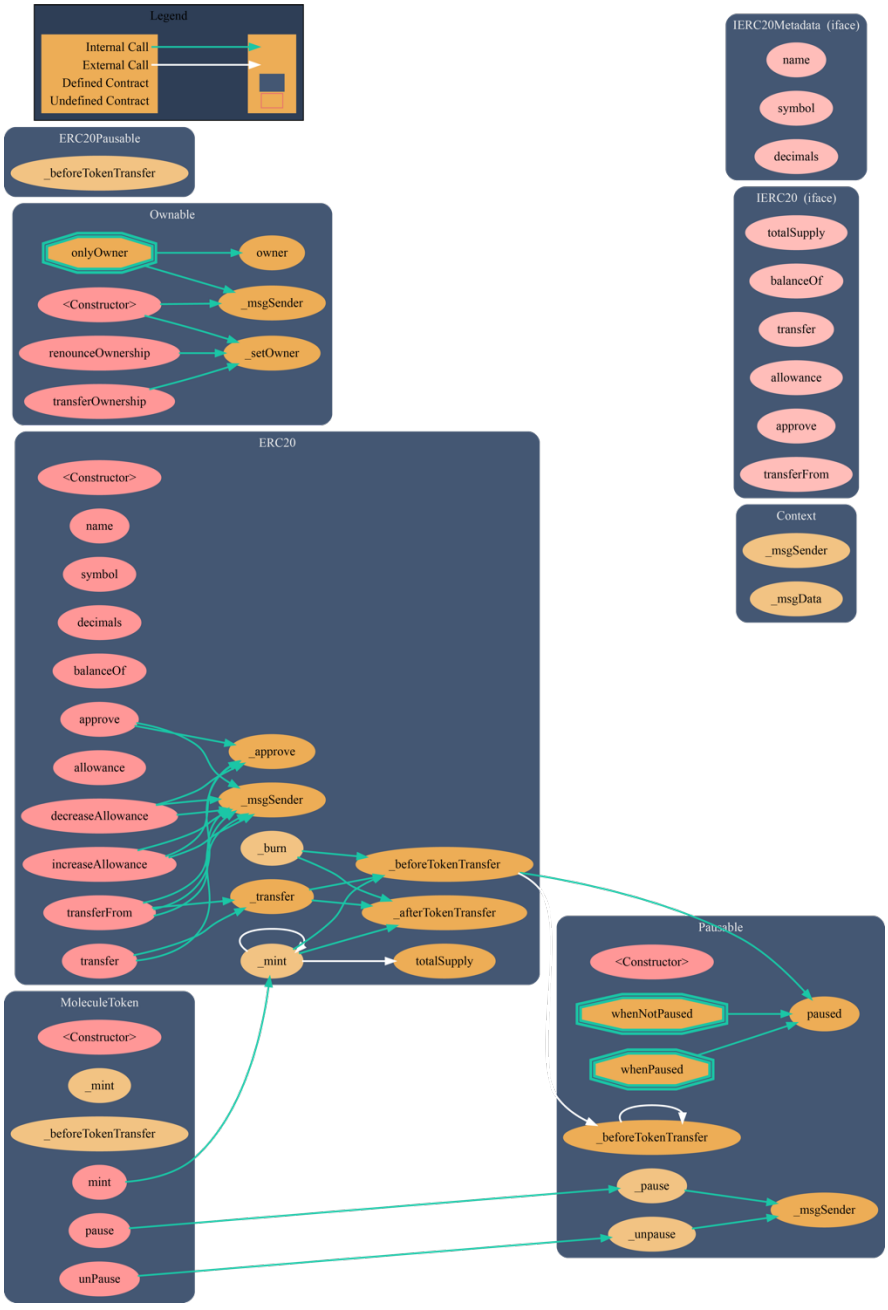

Image 1. Mollector Token Smart Contract call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Mar 20, 2021*	Public Report	TomoChainLab
1.1	*Mar 28, 2021*	Public Report	TomoChainLab

Table 3. Report versions history