

Audit Report April, 2024



For





Table of Content

Executive Summary	02
Number of Security Issues per Severity	03
Checked Vulnerabilities	04
Techniques and Methods	06
Types of Severity	07
Types of Issues	07
A. Contract - PioneersCollection.sol and PartnershipProcessor.sol	80
High Severity Issues	80
High Severity Issues Medium Severity Issues	08 08
Medium Severity Issues	08
Medium Severity Issues Low Severity Issues	08
Medium Severity Issues Low Severity Issues Informational Issues	08 08 08



Executive Summary

Project Name Quantelica Full Version (Merlin Chain)

Project URL https://quantelica.com/

Overview 1] The Pioneers Collection contract for minting and managing NFTs

from different subcollections.

2] PartnershipProcessor is a smart contract for handling

partnership operations with access control.

Audit Scope https://git.quantelica.com/blockchain/audit-271122/-/tree/full-version

Contracts in Scope Branch: Full-version

Contracts:-

-contracts/PartnershipProcessor.sol

-contracts/PioneersCollection.sol

and interfaces

Commit Hash f45b4346e82b9441997687d9147c008a898ef7ac

Language Solidity

Blockchain Merlin Chain

Method Manual Review, Automated Tools, Functional Testing

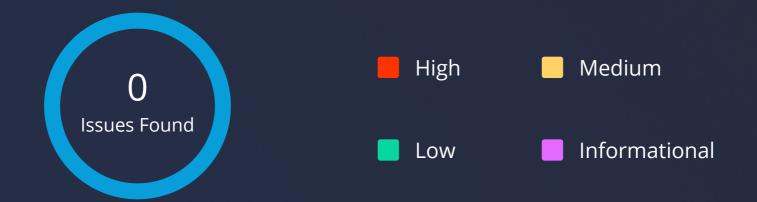
Review 1 12th April 2024 - 22nd April 2024

Updated Code Received 29th April 2024

Review 2 29th April 2024

Fixed In b81e4788c2839cf46b97303f1fa323d663de75f9

Number of Security Issues per Severity

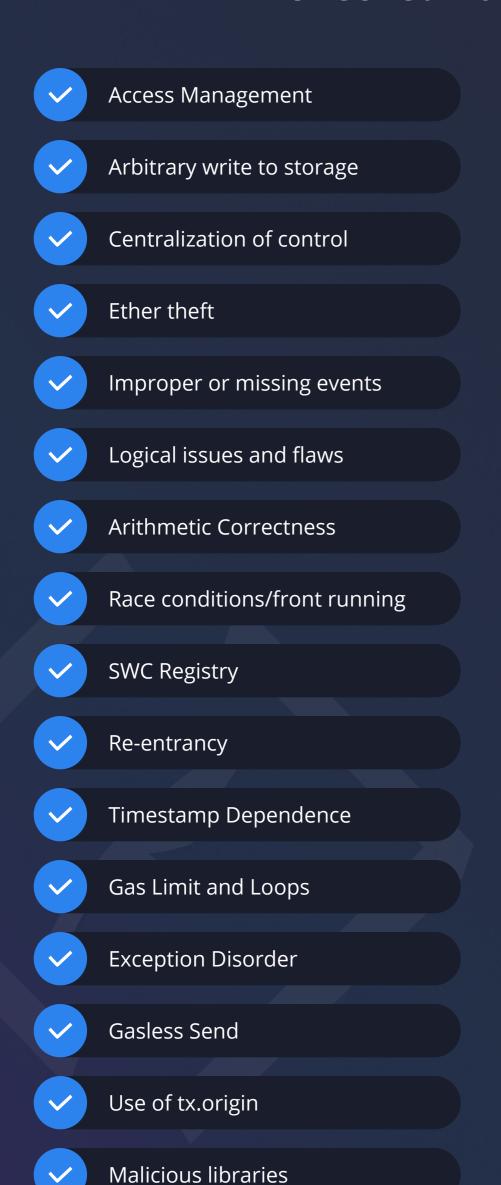


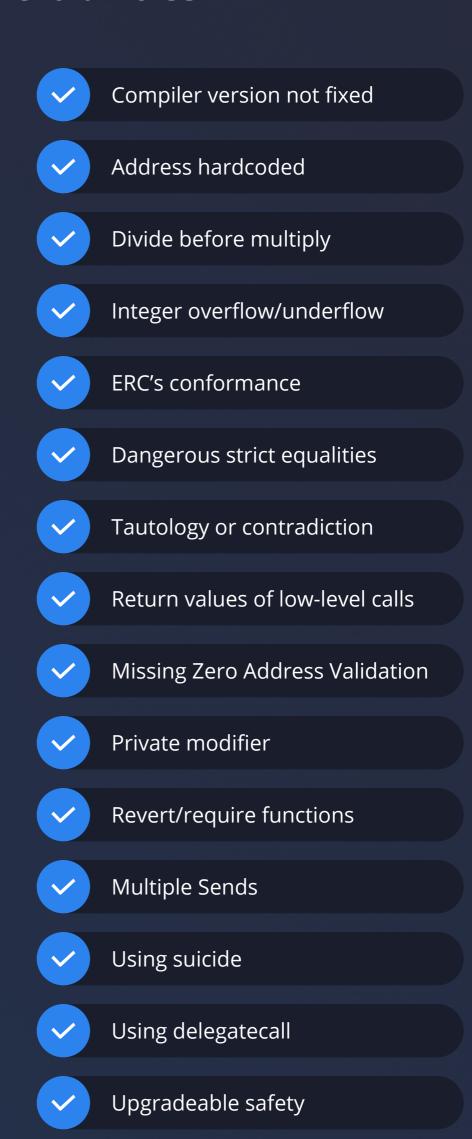
	High	Medium	Low	Informational
Open Issues	0	0	0	0
Acknowledged Issues	0	0	0	0
Partially Resolved Issues	0	0	0	0
Resolved Issues	0	0	0	0

Quantelica - Full Audit Report

www.quillaudits.com

Checked Vulnerabilities





Using throw



Checked Vulnerabilities

Using inline assembly

Style guide violation

Unsafe type inference

Implicit visibility level

Techniques and Methods

Throughout the audit of smart contracts, care was taken to ensure:

- The overall quality of code.
- Use of best practices.
- Code documentation and comments, match logic and expected behavior.
- Token distribution and calculations are as per the intended behavior mentioned in the whitepaper.
- Implementation of ERC standards.
- Efficient use of gas.
- Code is safe from re-entrancy and other vulnerabilities.

The following techniques, methods, and tools were used to review all the smart contracts.

Structural Analysis

In this step, we have analyzed the design patterns and structure of smart contracts. A thorough check was done to ensure the smart contract is structured in a way that will not result in future problems.

Static Analysis

A static Analysis of Smart Contracts was done to identify contract vulnerabilities. In this step, a series of automated tools are used to test the security of smart contracts.

Code Review / Manual Analysis

Manual Analysis or review of code was done to identify new vulnerabilities or verify the vulnerabilities found during the static analysis. Contracts were completely manually analyzed, their logic was checked and compared with the one described in the whitepaper. Besides, the results of the automated analysis were manually verified.

Gas Consumption

In this step, we have checked the behavior of smart contracts in production. Checks were done to know how much gas gets consumed and the possibilities of optimization of code to reduce gas consumption.

Tools and Platforms used for Audit

Remix IDE, Truffle, Solhint, Mythril, Slither, Solidity Static Analysis.



Quantelica - Full Audit Report

www.quillaudits.com 06

Types of Severity

Every issue in this report has been assigned to a severity level. There are four levels of severity, and each of them has been explained below.

High Severity Issues

A high severity issue or vulnerability means that your smart contract can be exploited. Issues on this level are critical to the smart contract's performance or functionality, and we recommend these issues be fixed before moving to a live environment.

Medium Severity Issues

The issues marked as medium severity usually arise because of errors and deficiencies in the smart contract code. Issues on this level could potentially bring problems, and they should still be fixed.

Low Severity Issues

Low-level severity issues can cause minor impact and are just warnings that can remain unfixed for now. It would be better to fix these issues at some point in the future.

Informational

These are four severity issues that indicate an improvement request, a general question, a cosmetic or documentation error, or a request for information. There is low-to-no impact.

Types of Issues

Open

Security vulnerabilities identified that must be resolved and are currently unresolved.

Resolved

These are the issues identified in the initial audit and have been successfully fixed.

Acknowledged

Vulnerabilities which have been acknowledged but are yet to be resolved.

Partially Resolved

Considerable efforts have been invested to reduce the risk/impact of the security issue, but are not completely resolved.

A. Contract - PioneersCollection.sol and PartnershipProcessor.sol

High Severity Issues

No issues were found.

Medium Severity Issues

No issues were found.

Low Severity Issues

No issues were found.

Informational Issues

No issues were found.

www.quillaudits.com 08

Automated Tests

No major issues were found. Some false positive errors were reported by the tools. All the other issues have been categorized above according to their level of severity.

Closing Summary

In this report, we have considered the security of the Quantelica Full version (Merlin Chain). We performed our audit according to the procedure described above.

The code is written nicely, and no issues are found.

Disclaimer

QuillAudits Smart contract security audit provides services to help identify and mitigate potential security risks in Quantelica Full version (Merlin Chain) smart contracts. However, it is important to understand that no security audit can guarantee complete protection against all possible security threats. QuillAudits audit reports are based on the information provided to us at the time of the audit, and we cannot guarantee the accuracy or completeness of this information. Additionally, the security landscape is constantly evolving, and new security threats may emerge after the audit has been completed.

Therefore, it is recommended that multiple audits and bug bounty programs be conducted to ensure the ongoing security of Quantelica Full version (Merlin Chain) smart contracts. One audit is not enough to guarantee complete protection against all possible security threats. It is important to implement proper risk management strategies and stay vigilant in monitoring your smart contracts for potential security risks.

QuillAudits cannot be held liable for any security breaches or losses that may occur subsequent to and despite using our audit services. It is the responsibility of the Quantelica Full version (Merlin Chain) to implement the recommendations provided in our audit reports and to take appropriate steps to mitigate potential security risks.

About QuillAudits

QuillAudits is a leading name in Web3 security, offering top-notch solutions to safeguard projects across DeFi, GameFi, NFT gaming, and all blockchain layers. With six years of expertise, we've secured over 1000 projects globally, averting over \$30 billion in losses. Our specialists rigorously audit smart contracts and ensure DApp safety on major platforms like Ethereum, BSC, Arbitrum, Algorand, Tron, Polygon, Polkadot, Fantom, NEAR, Solana, and others, guaranteeing your project's security with cutting-edge practices.



1000+ Audits Completed



\$30BSecured



1MLines of Code Audited



Follow Our Journey



















Audit Report April, 2024

For







- Canada, India, Singapore, UAE, UK
- www.quillaudits.com
- audits@quillhash.com