SECURITY AUDIT OF

PEGASUS GALAXY TOKEN AND VESTING SMART CONTRACTS

Public Report

Oct 21, 2021

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.
Polygon	Polygon is a protocol and a framework for building and connecting Ethereum-compatible blockchain networks. Aggregating scalable solutions on Ethereum supporting a multi-chain Ethereum ecosystem.
MATIC	A cryptocurrency whose blockchain is generated by the Polygon platform. Matic is used for payment of transactions and computing services in the Polygon 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 Verichains Lab on Oct 21, 2021. We would like to thank the Pegasus Galaxy Team 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 Pegasus Galaxy token and vesting smart contracts. 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 issues in the application.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About Pegasus Galaxy token and vesting smart contracts

Pegaxy is a play-to-earn PVP style horse racing game where players compete for top 3 placement against 11 other racers. Each race has randomised elemental variables which include wind, water, fire, speed and more. Using strategic upgrades, food and skill, players must place in the top 3 to earn the platforms utility token, VIS (Vigorus).

Within the game, players are able to breed, rent, sell, and of course race their Pega to earn VIS tokens. This system has proven to be a sound long-term economic approach when building an NFT/Blockchain based game as it enables teams to build large guilds, scholarship programs, and even provides solo players the opportunity to earn a second income through daily racing.

Pegasus Galaxy token contract is the ERC20 (TRC20) smart contract for Pegaxy's main token, Pegaxy Stone, or PGX in short. The vesting contract is smart contract to release vesting tokens uniformly within a period to investors/teams...

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of Pegasus Galaxy's token and vesting smart contracts. It was conducted on the source code provided by the Pegaxy 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

The initial review was conducted on Oct 18, 2021 and a total effort of 3 working days was dedicated to identifying and documenting security issues in the code base of the Pegasus Galaxy token and vesting smart contracts.

The following files were made available in the course of the review:

FILE	SHA256 SUM
PGXVesting.sol	937e3f0a4b5fc83ee5e8247b9201bd953f28ed4d49f88695993a526f3b56e690
Pegaxy.sol	0f5cf08771482f490ea0fe7d804e3913e5bb734c720980b455c57721a17794e4

2.2. Findings

The Pegasus Galaxy token and vesting smart contracts was written in Solidity language, with the required version to be ^0.8.0. The source code was written based on OpenZeppelin's library.

The audit team found no issue in the auditing contracts.

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*2021-10-18*	Public Report	Verichains Lab
1.1	*2021-10-21*	Public Report	Verichains Lab

Table 2. Report versions history