verichains

SECURITY AUDIT OF

NF3 AUTO COMPOUNDING SMART CONTRACT

Public Report

Dec 21, 2022

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.
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 was prepared by Verichains Lab on Dec 21, 2022. We would like to thank the NF3Labs 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 NF3 Auto Compounding Smart Contract. 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 vulnerable issue in the smart contracts code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About NF3 Auto Compounding Smart Contract

NF3 Auto Compounding Smart Contract is a staking vault where users can stake their assets (Fungible Tokens and Non-Fungible Tokens) and earn rewards.

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of the NF3 Auto Compounding Smart Contract.

It was conducted on commit `de19da0a5685e69417a321a9c9a49c600db97ba8` from git repository *https://github.com/NF3Labs/Auto-Compounding-Contract*.

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

SHA256 Sum	File
`6576cd1970768463053b4053d54a27165cf5d93f839cfd4c69c85f4d087d1a84`	`ERC4626.sol`
`abef5e936636efc0fcee710d18448ceff1d04328c5acb1a2783b26fe7c7ee5bb`	`ApeCoinStakeVault.sol`
`971484c445b36c4c52259ac6d0267c4e409a884674aa8a44b505cc8a3ed80502`	`ApeStaking.sol`

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 were 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 NF3 Auto Compounding Smart Contract contains two main contracts: `ApeCoinStakeVault` and `ApeStaking`. The source code was written based on the codebase of Graphes Staking contracts developed by Horizen Labs with some minor modifications.

2.1.1. ApeStaking contract

This contract contains four types of pools that can be staked to get rewards. These pools can be described below:

- **ApeCoin Pool**: As long as users have ApeCoin in their wallet, they can stake and start accruing rewards. No NFTs are required. For contract interaction purposes, the ApeCoin pool has a poolId of `0`.
- **BAYC Pool**: The BAYC pool allows users to stake up to a specific maximum amount of ApeCoin for each BAYC NFT they own. For contract interaction purposes, the BAYC pool has a poolId of `1`.
- **MAYC Pool**: The MAYC pool allows users to stake up to a specific maximum amount of ApeCoin for each MAYC NFT they own. For contract interaction purposes, the MAYC pool has a poolId of `2`.
- **Paired Pool (BAKC)**: The Paired pool works differently. A BAKC cannot stake ApeCoin alone and must be paired with a BAYC or MAYC. For contract interaction purposes, the Paired pool has a poolId of `3`.

2.1.2. ApeCoinStakeVault contract

This contract is an ERC4626 token contract which is a standard to optimize and unify the technical parameters of yield-bearing vaults. It provides a standard API for tokenized yield-bearing vaults that represent shares of a single underlying ERC-20 token. ERC-4626 also outlines an optional extension for tokenized vaults utilizing ERC-20, offering basic functionality for depositing, withdrawing tokens and reading balances.

The ApeCoin deposited to this contract will be staked into the ApeStaking contract to earn rewards.

2.2. Findings

During the audit process, the audit team found no vulnerability in the given version of NF3 Auto Compounding Smart Contract.

APPENDIX

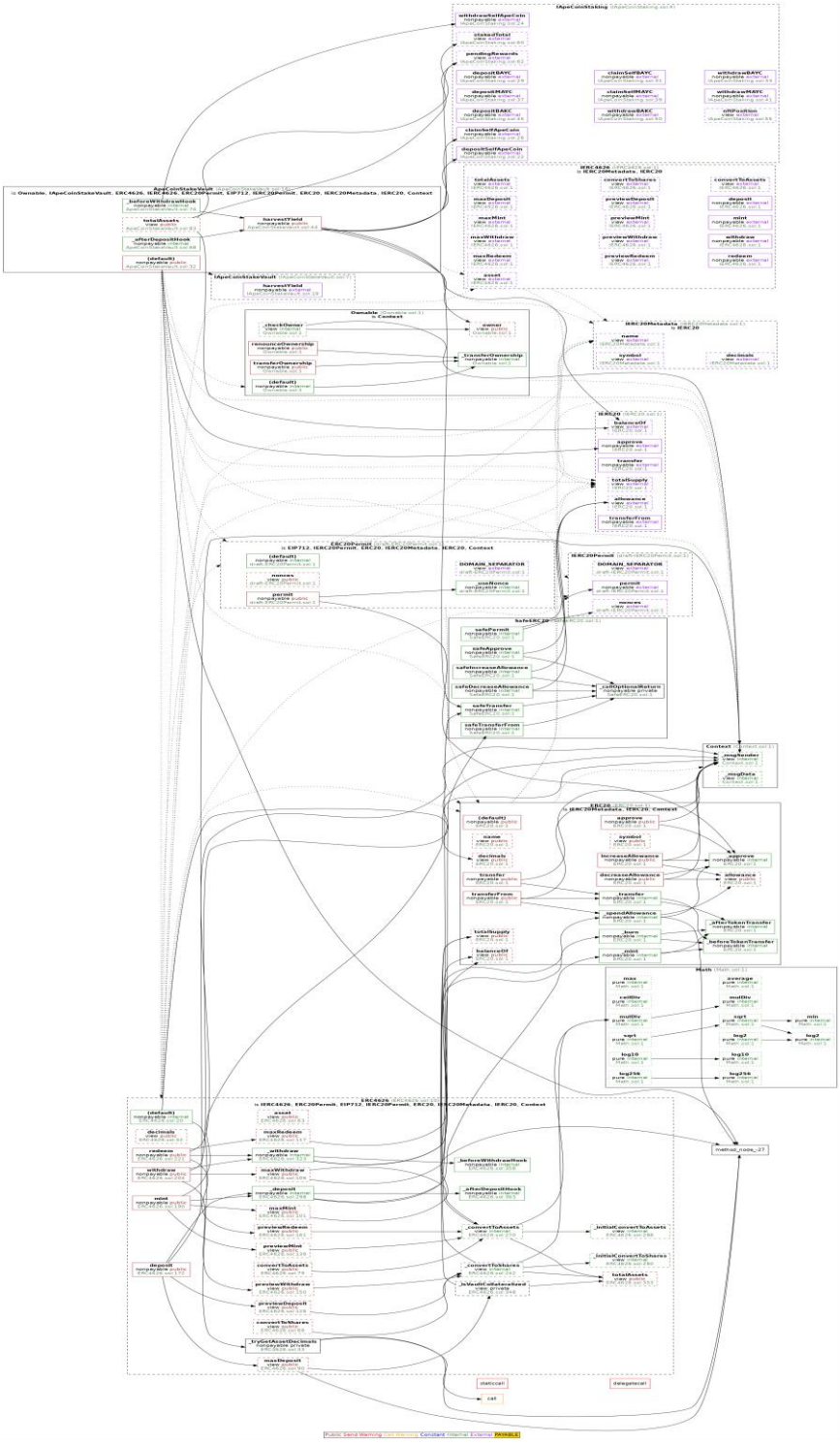

Image 1. ApeCoinStakeVault call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Dec 21, 2022*	Public Report	Verichains Lab

Table 2. Report versions history