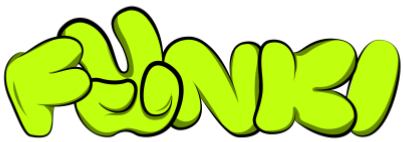
verichains

SECURITY AUDIT OF

FUNKI DEX V3 SMART CONTRACTS



Public Report

Oct 08, 2024

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 Oct 08, 2024. We would like to thank the Ather Labs 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 Funki DEX V3 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 vulnerable issues in the smart contracts code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY 5

1.1. About Funki DEX V3 Smart Contracts 5

1.2. Audit Scope 5

1.3. Audit Methodology 6

1.4. Disclaimer 7

1.5. Acceptance Minute 7

2. AUDIT RESULT 8

2.1. Overview 8

2.1.1. Core Contracts 8

2.1.2. Periphery Contracts 8

2.1.3. Swap Router Contracts 8

2.1.4. Staker Contracts 8

2.2. Findings 8

3. VERSION HISTORY 9

1. MANAGEMENT SUMMARY

1.1. About Funki DEX V3 Smart Contracts

Funki Dex V3 is a decentralized exchange (DEX) protocol built on the Ethereum blockchain. It introduces significant improvements over previous versions, including:

- **Concentrated Liquidity**: Liquidity providers (LPs) can allocate funds to specific price ranges, allowing them to earn fees more efficiently by concentrating capital where trading occurs.
- **Multiple Fee Tiers**: LPs can choose different fee tiers (0.05%, 0.30%, or 1%) based on the risk profile of the trading pair, providing more flexibility in managing risks.
- **Improved Capital Efficiency**: With concentrated liquidity, V3 is up to 4,000 times more capital efficient than V2, leading to lower slippage and better execution for traders.

1.2. Audit Scope

This audit focused on identifying security flaws in code and the design of the Funki DEX V3 Smart Contracts. It was conducted on commit `0a06288c1f2e2bba81f86f579d8fcb38704b4a41` from git repository link: *https://github.com/funkichain/dex-v3-contracts*.

The Funki DEX V3 Smart Contracts are forked from `Uniswap V3` protocol with the following repositories and commits:

Contract	Original Repository	Commit
v3-core	*https://github.com/funkichain/dex-v3-contracts*	0a06288c1f2e2bba81f86f579d8fcb38704b4a41
v3-periphery	*https://github.com/Uniswap/v3-periphery*	697c2474757ea89fec12a4e6db16a574fe259610
v3-staker	*https://github.com/Uniswap/v3-staker*	6d06fe4034e4eec53e1e587fc4770286466f4b35
swap-router-contracts	*https://github.com/Uniswap/swap-router-contracts*	0a06288c1f2e2bba81f86f579d8fcb38704b4a41

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

Ather Labs acknowledges that the security services provided by Verichains, are conducted to the best of their professional abilities but cannot guarantee 100% coverage of all security vulnerabilities. Ather Labs understands and accepts that despite rigorous auditing, certain vulnerabilities may remain undetected. Therefore, Ather Labs agrees that Verichains shall not be held responsible or liable, and shall not be charged for any hacking incidents that occur due to security vulnerabilities not identified during the audit process.

1.5. Acceptance Minute

This final report served by Verichains to the Ather Labs will be considered an Acceptance Minute. Within 7 days, if no any further responses or reports is received from the Ather Labs, the final report will be considered fully accepted by the Ather Labs without the signature.

2. AUDIT RESULT

2.1. Overview

The Funki DEX V3 Smart Contracts were written in `Solidity` language, with the required version to be `0.7.6`. They are forked from `Uniswap V3` protocol.

2.1.1. Core Contracts

The core consists of a single factory, a pool deployer, and the many pools the factory will create.

A significant amount of care and attention has been given to gas optimization in the core contracts. The result is a substantial reduction in gas costs for all protocol interactions compared to V2, at the cost of a reduction in code clarity.

2.1.2. Periphery Contracts

The periphery is a constellation of smart contracts designed to support domain-specific interactions with the core. As the Uniswap protocol is a permissionless system, the contracts described below have no special privileges and are only a small subset of possible periphery-like contracts.

2.1.3. Swap Router Contracts

The swap router supports all the basic requirements of a front-end offering trading. It natively supports single trades (x to y) and multihop trades (e.g. x to y to z).

2.1.4. Staker Contracts

The staker is a smart contract created to incentivize liquidity providers (LPs). Its primary function is to distribute rewards (usually in the form of tokens) to users who stake their liquidity positions in various pools.

2.2. Findings

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

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Oct 08, 2024*	Public Report	Verichains Lab

Table 2. Report versions history