AUDITS

SECURITY ASSESSMENT

SUPERNOVA

JUNE 03 RD 2022

AUDITS

TABLE OF CONTENTS

1 LEGAL DISCLAIMER

2 MH AUDITS INTRO

3 PROJECT SUMMARY

4 AUDIT SCORES

5 AUDIT SCOPE

6 METHODOLOGY

7 KEY FINDINGS

8 VULNERABILITIES

9 SOURCE CODE

10 APPENDIX

LEGAL DISCLAIMER

MH Audits are not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. These reports are not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts MH Audits to perform a security review.

MH Audits does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

MH Audits should not be used in any way to make decisions around investment or involvement with any particular project. These reports in no way provide investment advice, nor should be leveraged as investment advice of any sort.

MH Audits represents an extensive auditing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. MH Audits’ position is that each company and individual are responsible for their own due diligence and continuous security.

MH Audits’ goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

INTRODUCTION

MH Audits is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.

Secure your project with MH Audits

We offer field-proven audits with in-depth reporting and a range of suggestions to improve and avoid contract vulnerabilities.

Industry-leading comprehensive and transparent smart contract auditing on all public and private blockchains.

Vunerability checking

A crucial manual inspection carried out to eliminate any code flaws and security loopholes. This is vital to avoid vulnerabilities and exposures incurring costly errors at a later stage.

Contract verification

A thorough and comprehensive review in order to verify the safety of a smart contract and ensure it is ready for launch and built to protect the end-user.

Risk assessment

Analyse the architecture of the blockchain system to evaluate, assess and eliminate probable security breaches. This includes a full assessment of risk and a list of expert suggestions.

In-depth reporting

A truly custom exhaustive report that is transparent and depicts details of any identified threats and vulnerabilities and classifies those by severity.

Fast turnaround

We know that your time is valuable and therefore provide you with the fastest turnaround times in the industry to ensure that both your project and community are at ease.

Best-of-class blockchain engineers

Our engineers combine both experience and knowledge stemming from a large pool of developers at our disposal. We work with some of the brightest minds that have audited countless smart contracts over the last 4 years.

PROJECT SUMMARY

PROJECT INTRODUCTION

Supernova is a multi-platform (browser, mobile, PC) science-fiction open world MMORPG with Play-To-Earn feature, cryptocurrency, and NFT implementation. It is developed by the Sunday Games team since 2020. Supernova is going to be accessible by players with any experience both in blockchain and gaming.

Supernova will be a Free-To-Play game with open economics, no initial investment required. Players will be able to earn $LFC cryptocurrency in-game and sell or buy it from other players. $LFC is the only way to buy NFT Blueprints, earn Life Corporation gov tokens via staking, and more.

Project Name *Supernova*

Contract Name *LFC Token*

Contract Address *0xd9474595edb03e35c5843335f90eb18671921246*

Contract Chain *Mainnet*

Contract Type *Smart Contract*

Platform *EVM*

Language *Solidity*

Codebase *https://bscscan.com/address/0xd9474595edb03e35c5843335f90eb18671921246#code*

INFO & SOCIALS

Network *BNB Smart Chain (BEP20)*

Max Token Supply *100.000.000*

Website *https://sunday.games/supernova/*

Twitter *https://twitter.com/SupernovaVerse*

Telegram Chat *https://t.me/supernova_game*

Telegram Ann *https://t.me/supernova_game_news*

Discord *https://discord.com/4peUa4Xa3N*

Facebook *https://www.facebook.com/SupernovaVerse*

Youtube *https://www.youtube.com/c/SundayGamesStudio*

BSCScan *https://bscscan.com/token/0xd9474595edb03e35c5843335f90eb18671921246*

AUDIT SCORES

Issues	4
Critical	0
Major	0
Medium	0
Minor	2
Informational	2
Discussion	0

All issues are described in further detail on the following pages.

AUDIT SCOPE

FILE	LOCATION
LFCToken.sol	*contracts/LFCToken.sol*
Rewards.sol	*contracts/LFCToken.sol*

REVIEW METHODOLOGY

TECHNIQUES

This report has been prepared for Supernova to discover issues and vulnerabilities in the source code of the Supernova project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and SwiftScan Review techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts producedby industry leaders.

The security assessment resulted in findings that ranged from major to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective in the comments below.

TIMESTAMP

Version	v1.0
Date	2022/06/02
Description	Layout project Automated / Static security testing Summary
Version	v1.1
Date	2022/06/03
Description	Reaudit specific functions Partial manual review / Static security testing Final summary

KEY FINDINGS

TITLE	SEVERITY	STATUS
Outdated Complier Version	Minor	*Acknowledged*
Use of Floating Pragma	Minor	*Acknowledged*
Private Modifier Does Not Hide Data	Informational	*Acknowledged*
Redundant Fallback Reject	Informational	*Acknowledged*

Description:

Using an outdated compiler version can be problematic especially if there are publicly disclosed bugs and issues that affect the current compiler version. Compiler Version v0.8.13+commit.abaa5c0e

Location: *contracts/LFCToken.sol L02*

Issue: *Outdated Complier Version*

Level: *Minor*

Recommendation: *It is recommended to use a recent version of the Solidity compiler that should not be the most recent version, and it should not be an outdated version as well. Using very old versions of Solidity prevents the benefits of bug fixes and newer security checks. Consider using the solidity version* 0.8.4*, which patches most solidity vulnerabilities.*

Alleviation: *The Supernova team decided to stick to the current compiler version they utilize.*

Description:

Solidity source files indicate the versions of the compiler they can be compiled with using a pragma directive at the top of the solidity file. This can either be a floating pragma or a specific compiler version.

The contract was found to be using a floating pragma which is not considered safe as it can be compiled with all the versions described.

Location: *contracts/LFCToken.sol L02*

Issue: *Use of Floating Pragma*

Level: *Minor*

Recommendation: *It is recommended to follow the latter example, as future compiler versions may handle certain language constructions in a way the developer did not foresee. The developers should always use the exact Solidity compiler version when designing their contracts as it may break the changes in the future.*

pragma solidity ^0.4.17; not recommended -> compiles with 0.4.17 and above
pragma solidity 0.8.4; recommended -> compiles with 0.8.4 only

Alleviation: *The Supernova team took note of this exhibit.*

Description:

Everything that is inside a contract is visible to all observers external to the blockchain. Making something private *only prevents other contracts from reading or modifying the information, but it will still be visible to the whole world and observers of the blockchain.*

Miners have access to all contracts' code and data. Developers must account for the lack of privacy in Ethereum.

Location: *contracts/LFCToken.sol L22*

Issue: *Private Modifier Does Not Hide Data*

Level: *Informational*

Recommendation: *Keep in mind that the* private *modifier does not make a variable invisible and should not keep sensitive contents within the modifier.*

It is a best practice to use private *when you really want to protect your state variables and functions because you hide them behind logic executed through internal or public functions.*

Alleviation: *The Supernova team took note of this exhibit.*

Description:

The payment rejection fallback is redundant. Contracts should reject unexpected payments. Before Solidity 0.4.0, it was done manually: function () { revert(); }

Starting from Solidity 0.4.0, contracts without a fallback function automatically revert payments, making the code above redundant. Therefore, it is not required to have another function just to reject

Location: *contracts/LFCToken.sol L40~54*

Issue: *Redundant Fallback Reject*

Level: *Informational*

Recommendation: *Remove the function to save space: the contract will reject payments automatically. The fallback function can be used.*

Alleviation: *The Supernova team took note of this exhibit.*

https://bscscan.com/address/0xD9474595Edb03E35C5843335F90eb18671921246#code

FINDING CATEGORIES

The assessment process will utilize a mixture of static analysis, swift scan and other security techniques.

This report has been prepared for Supernova project using MH SwiftScan to examine and discover vulnerabilities and safe coding practices in Supernova's smart contract including the libraries used by the contract that are not officially recognized.

The scan runs a comprehensive static analysis on the solidity code and finds vulnerabilities ranging from minor gas optimizations to major vulnerabilities leading to the loss of funds. The coverage scope pays attention to all the informational and critical vulnerabilities with over (110+) modules. The scanning and auditing process covers the following areas:

Various common and uncommon attack vectors will be investigated to ensure that the smart contracts are secure from malicious actors. The scanner modules find and flag issues related to gas optimizations that help in reducing the overall gas cost It scans and evaluates the codebase against industry best practices and standards to ensure compliance It makes sure that the officially recognized libraries used in the code are secure and up to date.

AUDIT SCORES

MH Audits AuditScores is not a live dynamic score. It is a fixed value determined at the time of the report issuance date.

*Note that if no manual in-depth expert review has been performed a score multiplier of .9 will apply to the final result.

MH Audits AuditScores are not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. These reports and scores are not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts MH Audits to perform a security review.

AUDITS

WEBSITE
MHAUDITS.IO

TWITTER
@MHAUDITS