AUDITS

SECURITY ASSESSMENT

SEVN FINANCE

SEPTEMBER 26TH 2022

AUDITS

TABLE OF CONTENTS

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.

The report is provided only for the contract(s) mentioned in the report and does not include any other potential additions and/or contracts deployed by Owner. The report does not provide a review for contract(s), applications and/or operations, that are out of this report scope.

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.

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.

The security audit is not meant to replace functional testing done before a software release. As one audit-based assessment cannot be considered comprehensive, we always recommend proceeding with several independent manual audits and a public bug bounty program to ensure the security of the smart contracts.

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

Sevn Finance is the first multichain gaming DeFi protocol. It combined the best practices of decentralized exchanges (DEX) and GameFi orientation. Multichain DEX - trade tokens on-chain; Yield farming - Provide liquidity and earn profit; NFT rental marketplace - rent the best NFTs and play P2E games.

Sevn Finance uses the veTokenomics model to maximize TVL and minimize selling pressure. SEVN holders can lock their SEVN in the holder pool to receive veSEVN. Dividents - SEVN lets you earn part of the protocol profit; Governance - SENV has a voting power that allows users to participate in governance; Boosted Farms - SEVN gives boosted rewards on farms of up to 2,5x.

Project Name *Sevn Finance*

Contract Name *SEVN*

Contract Address -

Contract Chain *Not Yet Deployed On Mainnet*

Contract Type *Smart Contract*

Platform *EVM*

Language *Solidity*

Codebase *GitHub Repository*

INFO & SOCIALS

Network *Ethereum (ERC20)*

Total Supply *350,000,000*

Website *https://www.sevn.finance/*

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

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

Telegram Ann -

Discord *https://discord.gg/PZD2BFkRAj*

Medium *https://medium.com/@SevnFinance*

GitHub *https://github.com/Sevn-finance*

EtherScan -

AUDIT SCORES

Issues	3
Critical	0
Major	0
Medium	0
Minor	0
Informational	3
Discussion	0

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

AUDIT SCOPE

FILE	LOCATION
Sevn.sol	*GitHub Repository sevn-finance/sevn-token/contracts/*
Mintable.sol	*GitHub Repository sevn-finance/sevn-token/contracts/library/*

REVIEW METHODOLOGY

TECHNIQUES

This report has been prepared for Sevn Finance to discover issues and vulnerabilities in the source code of the Sevn Finance project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Dynamic, Static Analysis and Manual 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.
- Thorough line-by-line manual review of the entire codebase by industry experts.

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	2022/09/19
Description	Layout project Automated / Manual review / Static & dynamic security testing Summary
Version	v1.1
Date	2022/09/26
Description	Reaudit addressed issues Final summary

KEY FINDINGS

TITLE	SEVERITY	STATUS
Functions Should Be Declared External	◆ **Gas**	*Fixed*
Missing NatSpec Comments	◆ **Informational**	*Fixed*
Large Number Literals	◆ **Gas**	*Acknowledged*

Description: *Public functions that are never called by a contract should be declared external in order to conserve gas.*

The following functions were declared as public but were not called anywhere in the contract, making the public visibility useless.

Affected Code: mint() - ***L18-L22***
burn() - ***L24-L26***
addMinter() - ***L28-L31***
delMinter() - ***L33-L36***

Impacts: *Smart Contracts are required to have effective Gas usage as they cost real money, and each function should be monitored for the amount of gas it costs to make it gas efficient.*

public ***functions cost more Gas than*** external ***functions.***

Issue: *Functions Should Be Declared* External

Type: *Gas Optimization*

Level: *Gas*

Remediation: *Use the* external *state visibility for functions that are never called from inside the contract.*

Alleviation / Retest: *All the required public functions visibility has been set to* external.

Description: *Solidity contracts use a special form of comments to document code. This special form is named the Ethereum Natural Language Specification Format (NatSpec).*

The document is divided into descriptions for developers and end-users along with the title and the author.

The contract Sevn.sol *is missing NatSpec comments in the code which makes it difficult for the auditors and future developers to understand the code.*

Affected Code: *Sevn.sol*
Mintable.sol

Impacts: *Missing NatSpec comments and documentation about a library or a contract affect the audit and future development of the smart contracts.*

Issue: *Missing NatSpec Comments*

Type: *Missing Best Practices*

Level: *Informational*

Remediation: *Add necessary NatSpec comments inside the library along with documentation specifying what it's for and how it's implemented.*

Alleviation / Retest: *NatSpec comments have been added properly.*

Description: *Solidity supports multiple rational and integer literals, including decimal fractions and scientific notations. The use of very large numbers with too many digits was detected in the code that could have been optimized using a different notation also supported by Solidity.*

Affected Code: *Sevn.sol L10-L11*

```
  uint256 public constant preMineSupply = 45350000 * 1e18; // 45 350 000
  uint256 public constant maxSupply = 350000000 * 1e18; // 350 000 000
```

Impacts: *Having a large number literals in the code increases the gas usage of the contract while its deployment and when the functions are used or called from the contract. It also makes the code harder to read and audit and increases the chances of introducing code errors.*

Issue: *Large Number Literals*

Type: *Gas & Missing Best Practices*

Level: *Gas*

Remediation: *Scientific notation in the form of 2e10 is also supported, where the mantissa can be fractional but the exponent has to be an integer. The literal MeE is equivalent to M * 10**E. Examples include 2e10, 2e10, 2e-10, 2.5e1, as suggested in official solidity documentation.*

https://docs.soliditylang.org/en/latest/types.html#rational-and-integer-literals

It is recommended to use numbers in the form 35 * 1e7 * 1e18 ***or*** 35 * 1e25***. The numbers can also be represented by using underscores between them to make them more readable such as*** 35_00_00_000

Alleviation / Retest: *The team has decided to leave this as it is because the numbers won't change and hence the chances of errors are negligible.*

https://github.com/Sevn-finance/sevn-token

FINDING CATEGORIES

The assessment process will utilize a mixture of static analysis, dynamic analysis, in-depth manual review and/or other security techniques.

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

A comprehensive static and dynamic analysis has been performed on the solidity code in order to find vulnerabilities ranging from minor gas optimizations to major vulnerabilities leading to the loss of funds.

Various common and uncommon attack vectors will be investigated to ensure that the smart contracts are secure from malicious actors. The testing methods 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.

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