HACKEN

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Customer: ACHIP & ACHAIR GUILD VENTURES PTE. LTD.
Date: March 15th, 2022

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Document

Name	Smart Contract Code Review and Security Analysis Report for ACHIP & ACHAIR GUILD VENTURES PTE. LTD.
Approved by	Andrew Matiukhin | CTO Hacken OU
Type	Staking; ERC721 token
Platform	Harmony / Solidity
Methods	Architecture Review, Functional Testing, Computer-Aided Verification, Manual Review
Repositories	https://github.com/AAG-Ventures/aag-staking-contract https://github.com/AAG-Ventures/aag-genesis-nft-contract
Commits	af2f17ea2b8ea046398703f8b061210c821eba09 fe877e1cc7e6386602f5d1c07e5c937f0e8f5f80
Deployed contract	–
Technical Documentation	YES
JS tests	YES
Website	https://aag.ventures
Timeline	14 FEBRUARY 2022 - 15 MARCH 2022
Changelog	18 FEBRUARY 2022 - Initial Audit 01 MARCH 2022 - Second Review 15 MARCH 2022 - Third Review

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Table of contents

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Introduction

Hacken OÜ (Consultant) was contracted by ACHIP & ACHAIR GUILD VENTURES PTE. LTD. (Customer) to conduct a Smart Contract Code Review and Security Analysis. This report presents the findings of the security assessment of the Customer's smart contract and its code review conducted between February 14th, 2022 - March 15t, 2022.

Scope

The scope of the project is smart contracts in two repositories:

Repository 1:
https://github.com/AAG-Ventures/aag-staking-contract
Commit 1:
af2f17ea2b8ea046398703f8b061210c821eba09
Repository 2:
https://github.com/AAG-Ventures/aag-genesis-nft-contract
Commit 2:
fe877e1cc7e6386602f5d1c07e5c937f0e8f5f80
Technical Documentation: Yes
- Staking Litepaper:
  https://litepaper.aag.ventures/the-usdaag-token/staking
- Litepaper:
  https://litepaper.aag.ventures/

JS tests: Yes
- Staking:
  https://github.com/AAG-Ventures/aag-staking-contract/tree/af2f17ea2b8ea046398703f8b061210c821eba09/tests
- NFT:
  https://github.com/AAG-Ventures/aag-genesis-nft-contract/tree/fe877e1cc7e6386602f5d1c07e5c937f0e8f5f80/tests

Contracts:
Staking:
core/ReceiptCore.sol
core/ReceiptEnum.sol
core/ShareCore.sol
interfaces/INFT.sol
interfaces/IReceiptCore.sol
interfaces/IStake.sol
upgrade/StakingAdmin.sol
upgrade/StakingProxy.sol
LPStakeV1.sol
SingleStakeV1.sol
NFT:
core/CoreNFT.sol
interface
interface/IAAGNFT.sol
interface/IStake.sol
upgrade/NFTAdmin.sol
upgrade/NFTProxy.sol
AAGNFTV3.sol

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

We have scanned this smart contract for commonly known and more specific vulnerabilities. Here are some of the commonly known vulnerabilities that are considered:

Category	Check Item
Code review	▪ Reentrancy ▪ Ownership Takeover ▪ Timestamp Dependence ▪ Gas Limit and Loops ▪ DoS with (Unexpected) Throw ▪ DoS with Block Gas Limit ▪ Transaction-Ordering Dependence ▪ Style guide violation ▪ Costly Loop ▪ ERC20 API violation ▪ Unchecked external call ▪ Unchecked math ▪ Unsafe type inference ▪ Implicit visibility level ▪ Deployment Consistency ▪ Repository Consistency ▪ Data Consistency
Functional review	▪ Business Logics Review ▪ Functionality Checks ▪ Access Control & Authorization ▪ Escrow manipulation ▪ Token Supply manipulation ▪ Assets integrity ▪ User Balances manipulation ▪ Data Consistency manipulation ▪ Kill-Switch Mechanism ▪ Operation Trails & Event Generation

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Executive Summary

According to the assessment, the Customer's smart contracts are poor-secured.

Our team analyzed code functionality, manual audit, and automated checks with Mythril and Slither. All issues found during automated analysis were manually reviewed, and important vulnerabilities are presented in the Audit overview section. All found issues can be found in the Audit overview section.

As a result of the audit, security engineers found **1** high, **2** medium, and **3** low severity issues.

After the second review, security engineers found **no security issue**.

After the third review, security engineers found that in the contract there were introduced min/max staking token lock duration functionality and

min reward token lock duration. Therefore **no security issues** were found.

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to assets loss or data manipulations.
High	High-level vulnerabilities are difficult to exploit; however, they also have a significant impact on smart contract execution, e.g., public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to assets loss or data manipulations.
Low	Low-level vulnerabilities are mostly related to outdated, unused, etc. code snippets that can't have a significant impact on execution

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Audit overview

Critical

No critical issues were found.

High

1. Unprotected initializer functions.

   Contracts have an initializer function that is not protected by ownership checking. That means that the attacker could spy on the network to catch the contract creation call and send initialize function call from their own with a higher gas price to take control over the staking contract.

   Contracts: SingleStake, LPStake, AAGNFTV3

   Functions: SingleStake.initialize, LPStake.initialize, AAGNFTV3.init

Recommendation: Please add the “onlyOwner” modifier to the initializer functions.

Status: Not related. The initializer is called right after the instantiating in the Proxy contract.

Medium

1. The test could not be run.

   Tests in the NFT scope could not be run because of a typo: trying to import AAGNFTV1 from “dist/types” whenever there is only the AAGNFTV3 contract.

   Scope: NFT Tests

Recommendation: Please fix the issue.

Status: Fixed

2. Very low test coverage.

   Test coverage is about 36% for statements and 13% for code branches which is too low.

   Scope: NFT Tests

Recommendation: Please make sure test coverage is at least 95% for statements and up to 100% for branches.

Status: Fixed

Low

1. No implementation checked.

   Any contract address could be set as the NFT token or Single/LP Stake for contracts by mistake. No interface implementation check is added.

   Contracts: SingleStake, LPStake, AAGNFTV3

   Functions: SingleStake.setNFT, LPStake.setNFT, AAGNFTV3.setSingleStake, AAGNFTV3.setLPStake

Recommendation: Please consider the EIP165 usage.

Status: Fixed

2. Boolean equality.

   Boolean constants can be used directly and do not need to be compared to **true** or **false**.

   Contract: AAGNFTV3

   Functions: setLock, setUnlock

Recommendation: Remove the equality to the boolean constant.

Status: Fixed

3. Unused private state variable.

   The state variable “_counter” is never used in the AAGNFTV3 contract.

   Contract: AAGNFTV3

   Variable: _counter

Recommendation: Remove unused state variables.

Status: Fixed

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Conclusion

Smart contracts within the scope were manually reviewed and analyzed with static analysis tools.

The audit report contains all found security vulnerabilities and other issues in the reviewed code.

As a result of the audit, security engineers found **1** high, **2** medium, and **3** low severity issues.

After the second review, security engineers found **no security issue**.

After the third review, security engineers found that in the contract, there were introduced min/max staking token lock duration functionality and min reward token lock duration. Therefore **no security issues** were found.

Hacken OÜ
Parda 4, Kesklinn, Tallinn,
10151 Harju Maakond, Eesti,
Kesklinna, Estonia
support@hacken.io

Disclaimers

Hacken Disclaimer

The smart contracts given for audit have been analyzed by the best industry practices at the date of this report, with cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functions).

The audit makes no statements or warranties on the security of the code. It also cannot be considered a sufficient assessment regarding the utility and safety of the code, bug-free status, or any other contract statements. While we have done our best in conducting the analysis and producing this report, it is important to note that it should not rely on this report only – we recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts.

Technical Disclaimer

Smart contracts are deployed and executed on a blockchain platform. The platform, its programming language, and other software related to the smart contract can have vulnerabilities that can lead to hacks. Thus, the audit can not guarantee the explicit security of the audited smart contracts.