Staking - galactic-dao-contracts - Report

Prepared for Galactic Punks, 17 March 2022

Table of Contents

Document control

Document changes

Version	Date	Name	Changes
0.1	2022-03-14	Vinicius Marino	Initial report
0.2	2022-03-15	Vinicius Marino	Team communication and Pre-Release
1.0	2022-03-17	Vinicius Marino	Final report release

Document contributors

Name	Role	Email address
Vinicius Marino	Security Specialist	vini@scv.services

Introduction

SCV was engaged by Galactic Punks to assist in identifying security threats and vulnerabilities that have the potential to affect their security posture. Additionally, SCV will assist the team in understanding the risks and identifying potential mitigations.

Scope

SCV performed the security assessment on the following `Staking` contract.

- https://github.com/galactic-dao/galactic-dao-contracts/8861595f82a99fc5b5abf2664fac4b409bf3fa0f

Vulnerabilities were remediated by Galactic Punks team in the following commit hash:

- https://github.com/galactic-dao/galactic-dao-contracts/7ae9886a5cddd9afeca9167e61c196043f4e0e32

Methodologies

SCV performs a combination of automated and manual security testing based on the scope of testing. The testing performed is based on the extensive experience and knowledge of the auditor to provide the greatest coverage and value to Galactic Punks. Testing includes, but is not limited to, the following:

- Understanding the application and its code base purpose;
- Deploying SCV in-house tooling to automate dependency analysis and static code review;
- Analysis each line of the code base and inspect application perimeter;
- Review underlying infrastructure technologies and supply chain security posture;

Code Criteria and Test Coverage

SCV is using a scale from **0** to **10** that represents how `SUFFICIENT(6-10)` or `NOT SUFFICIENT(0-5)` each code criteria was assessed:

Criteria	Status	Scale Range	Notes
Provided Documentation	**Not Sufficient**	1-2	N\A
Code Coverage Test	**Sufficient**	6-7	N\A
Code Readability	**Sufficient**	7-8	N\A
Code Complexity	**Sufficient**	5-6	N\A

Vulnerabilities Summary

	Title and Summary	Risk	Status
1	Contract might run out-of-gas due storage size grow	**Low**	**Remediated**
2	Lack of validations on execute_change_config() storage updates	**Informational**	**Remediated**

Detailed Vulnerabilities

Vulnerability 1: Contract might run out-of-gas due storage size grow

Likelihood	Impact	Risk
Possible	Low	**Low**

Description

The `token_distributions()` handles may grow arbitrarily large in size, as a result, it can lead to rewards getting locked away indefinitely caused by out of gas errors.

Recommendations

SCV suggest maintaining cumulative sum of rewards per token, instead of a list of rewards.

A technical way to describe how this can be implemented:

1. we always maintain an array `A` such that `A[t]` is the cumulative sum of rewards for token `t`;
2. when token `t` is received, we compute `amount_per_stake`, and increment `A[t]` by `amount_per_stake`;
3. when you stake a NFT, we can take a snapshot of `A`, and record it;
4. when we claim rewards/withdraw, the amount of rewards returned is `A_current` – `A_snapshot`, and then we update `A_snapshot` to `A_current`.

This approach above would be much cleaner, and doesn't involve dealing with complicated indexes and ranges.

Vulnerability 2: Lack of validations on execute_change_config() storage updates

Likelihood	Impact	Risk
Rare	Informational	**Informational**

Description

There is no validation on `execute_change_config()` function when executed by the contract admin. An human error, like a “typo” or a wrong copy & past might cause the contract to panic.

Recommendations

Enforce validation on all configurable attributes.

Appendices

Appendix A: Report Disclaimer

Appendix B: Risk assessment methodology

A qualitative risk assessment is performed on each vulnerability to determine the impact and likelihood of each.

Risk rate will be calculated on a scale. As per criteria Likelihood vs Impact table below:

Impact \ Likelihood	Rare	Unlikely	Possible	Likely
Critical	**Medium**	**High**	**Critical**	**Critical**
Severe	**Low**	**Medium**	**High**	**High**
Moderate	**Low**	**Medium**	**Medium**	**High**
Low	**Low**	**Low**	**Low**	**Medium**
Informational	**Informational**	**Informational**	**Informational**	**Informational**

LIKELIHOOD:

- **Likely**: likely a security incident will occur;
- **Possible**: It is possible a security incident can occur;
- **Unlikely**: Low probability a security incident will occur;
- **Rare**: In rare situations, a security incident can occur;

IMPACT:

- **Critical**: May cause a significant and critical impact;
- **Severe**: May cause a severe impact;
- **Moderate**: May cause a moderated impact;
- **Low**: May cause low or none impact;
- **Informational**: May cause very low impact or none.

SCV