Smart Contract Audit

Sovryn Staking

Franklin Richards

27th January 2021

Index

[Index](#)

[Introduction](#)

[High Threats](#)

[Medium Threats](#)

[Low Threats](#)

[Optimization & Readability](#)

[Typo’s & Comments](#)

[Limitations](#)

[Ambiguity](#)

[Suggestions](#)

Note: Some threat levels or headings might be empty if there is no vulnerability/updates/suggestions found.

Introduction

The contract audited here and the supporting contracts is/are:
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/7f43579/contracts/governance/Staking/Checkpoints.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/7f43579/contracts/governance/Staking/IStaking.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/5b41878/contracts/governance/Staking/SafeMath96.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/b76b3ba/contracts/governance/Staking/Staking.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/8b54d07/contracts/governance/Staking/StakingProxy.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/7f43579/contracts/governance/Staking/StakingStorage.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/13833fc/contracts/governance/Staking/WeightedStaking.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/a95abd4/contracts/governance/Vesting/IVesting.sol
https://github.com/DistributedCollective/Sovryn-smart-contracts/blob/7f43579/contracts/governance/IFeeSharingProxy.sol
and was shared by Sovryn for auditing purposes while working with them.

Based on the audit, we were able to find 0 High threats, 0 Medium threat, and 0 Low threat with some other changes in the optimization, readability, typos, and comments section.

High Threats

1. None

Medium Threats

1. None

Low Threats

1. None

Optimization & Readability

1. The naming `latest` (used multiple times in the contract) sounds a bit odd, considering it represents the maximum duration in the future until which the staking is possible. Different choices of words reflecting the same can aid in readability.
2. Instead of reading `oldStake` in _writeDelegateCheckpoint() from storage, we can introduce a function parameter, which passes the oldStake from the called functions (_increaseDelegateStake() and _decreaseDelegateStake()).
3. A descriptive error message is always great for debugging as well as in production. Change required in Line 164.

Typo's & Comments

1. Error message not mentioned for Line 96.
2. Please remove unnecessary comments like in Line 72, Line 139, etc.

Limitations

1. Currently, if we are staking until a particular time, say XYZ, which is a considerably long time. If that person wants to divide his stake delegation to more than 1 person/address, he cannot do that. Ex: Like if a person stakes 100K Tokens and want to provide Person A with 30% or 30K of Token Delegation, Person B with 50% or 50K of Token Delegation and Person C with 20% or 20K of Token Delegation, until a certain time, it is not possible. The workaround is to choose different until dates (which should be possible through Sovryn interface as well).

Suggestions

1. Removal of migrateToNewStakingContract().