Smart Contract Security Audit Report

Gearbox Governor

1. Contents

2. General Information

This report contains information about the results of the security audit of the Gearbox (hereafter referred to as "Customer") Governor smart contract, conducted by Decurity in the period from 11/08/2023 to 11/20/2023 (including remediation verification).

2.1. Introduction

Tasks solved during the work are:

- Review the protocol design and the usage of 3rd party dependencies,
- Audit the contracts implementation,
- Develop the recommendations and suggestions to improve the security of the contracts.

2.2. Scope of Work

The audit scope included the contract Governor (GovernorV3) in the following repository: https://github.com/Gearbox-protocol/governance. Initial review was done for the commit cdefd21dc58c4069c4ca716b89a5c9f258875bce and the re-testing was done for the commit c90434702c163f3f1c2cb4db90cece525160ee07.

2.3. Threat Model

The assessment presumes the actions of an intruder who might have the capabilities of any role. The main possible threat actors are:

- Arbitrary user,
- Queue admin,
- Veto admin,
- Timelock contract.

2.4. Weakness Scoring

An expert evaluation scores the findings in this report, and the impact of each vulnerability is calculated based on its ease of exploitation (based on the industry practice and our experience) and severity (for the considered threats).

2.5. Disclaimer

Due to the intrinsic nature of the software and vulnerabilities and the changing threat landscape, it cannot be generally guaranteed that a certain security property of a program holds.

Therefore, this report is provided “as is” and is not a guarantee that the analyzed system does not contain any other security weaknesses or vulnerabilities. Furthermore, this report is not an endorsement of the Customer’s project, nor is it an investment advice.

That being said, Decurity exercises the best effort to perform its contractual obligations and follow the industry methodologies to discover as many weaknesses as possible and maximize the audit coverage using limited resources.

3. Summary

As a result of this work, we have not discovered any exploitable security issues.

The other suggestions included some low-risk issues or potential weaknesses which has been fixed and re-tested in the course of the work.

The Gearbox team has given feedback for the suggested changes and an explanation for the underlying code.

3.1. Suggestions

The table below contains the discovered issues, their risk level, and their status as of May 3, 2023.

Table. Discovered weaknesses

Issue	Contract q	Risk Level	Status
Veto admin can prevent changing of the veto admin	Governor.sol	Low	Acknowledged
Malicious queue admins can back-run the batches	Governor.sol	Low	Fixed
Lack of address(0) check	Governor.sol	Low	Fixed
Batch ETA equals the maximum of each transaction's ETA	Governor.sol	Info	Fixed

4. General Recommendations

This section contains general recommendations on how to improve the overall security level.

The Findings section contains technical recommendations for each discovered issue.

4.1. Security Process Improvement

The following is a brief long-term action plan to mitigate further weaknesses and bring the product security to a higher level:

- Keep the whitepaper and documentation updated to make it consistent with the implementation and the intended use cases of the system,
- Perform regular audits for all the new contracts and updates,
- Ensure the secure off-chain storage and processing of the credentials (e.g. the privileged private keys),
- Launch a public bug bounty campaign for the contracts.

5. Findings

5.1. Veto admin can prevent changing of the veto admin

Risk Level: Low

Status: The Customer noted that the veto admin address is supposed to be a highly trusted multisig and they don't want to add overly complex checks.

Description:

A malicious veto admin can potentially prevent further updates to the veto admin address.

Remediation:

Prevent transaction cancelling if the transaction to be cancelled is a call to `updateVetoAdmin` in the current contract.

5.2. Malicious queue admins can back-run the batches

Risk Level: Low

Status: Fixed in the commit d9fd78c6. The batch initiator check has been added.

Description:

A malicious queue admin can add a transaction to a batch by back-running the batch creation. This can go unnoticed since the additional unverified transaction was not expected to be in the batch.

Remediation:

Add the authorization check in the `queueTransaction` function to verify that the batch has been started by the sender.

5.3. Lack of address(0) check

Risk Level: Low

Status: Fixed in the commit d9fd78c6. The queue and veto admin addresses are now checked.

Description:

Several functions lack address(0) check for input variables.

```
L192: addQueueAdmin(address _admin)
L212: updateVetoAdmin(address _admin)
```

Remediation:

Consider adding address(0) checks to the listed functions.

5.4. Batch ETA equals the maximum of each transaction's ETA

Risk Level: Info

Status: Fixed in the commit d9fd78c6. The batch ETA parameter has been introduced.

Description:

If at the time of batch execution ETA of one of the transactions hasn't been reached, the whole batch will revert.

Therefore, the cumulative ETA of a batch is equal to the highest ETA in the batch which could be unintended.

Remediation:

Set ETA for the batches as a single parameter.

6. Appendix

6.1. About us

The Decurity team consists of experienced hackers who have been doing application security assessments and penetration testing for over a decade.

During the recent years, we've gained expertise in the blockchain field and have conducted numerous audits for both centralized and decentralized projects: exchanges, protocols, and blockchain nodes.

Our efforts have helped to protect hundreds of millions of dollars and make web3 a safer place.