Smart Contract Security Audit Report

Gearbox Infiniti & Uniswap v4 Integrations

1. Contents

2. General Information

This report contains information about the results of the security audit of the Gearbox (hereafter referred to as "Customer") smart contracts, conducted by Decurity in the period from 2025-08-26 to 2025-09-04.

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 contracts in the following repository: https://github.com/Gearbox-protocol/integrations-v3. Initial review was done for the commit 29bcf3c4adc90ff6b125887a58c8f08beaf6c2bf and the re-testing was done for the commit b3cc54453225b0f163aaa48f812dbd5ff5c9a148.

The following contracts have been tested:

- adapters/infinifi/InfinifiGatewayAdapter.sol
- adapters/infinifi/InfinifiUnwindingGatewayAdapter.sol
- helpers/infinifi/InfinifiUnwindingPhantomToken.sol
- helpers/infinifi/InfinifiUnwindingGateway.sol
- adapters/uniswap/UniswapV4.sol
- helpers/uniswap/UniswapV4Gateway.sol

2.3. Threat Model

The assessment presumes actions of an intruder who might have capabilities of any role (an external user, token owner, token service owner, a contract). The centralization risks have not been considered upon the request of the Customer.

The main possible threat actors are:

- User,
- Protocol owner,
- Liquidity Token owner/contract.

2.4. Weakness Scoring

An expert evaluation scores the findings in this report, an 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 best effort to perform their contractual obligations and follow the industry methodologies to discover as many weaknesses as possible and maximize the audit coverage using the limited resources.

3. Summary

As a result of this work, we have not discovered any critical exploitable security issues. The other suggestions included fixing the low-risk issues and some best practices.

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

3.1. Suggestions

The table below contains the discovered issues, their risk level, and their status as of November 25, 2025.

Table. Discovered weaknesses

Issue	Contract	Risk Level	Status
setLockedTokenBatchStatus doesn't remove tokens from unwindingEpochToLockedToken	adapters/infinifi/InfinifiGatewayAdapter.sol adapters/infinifi/InfinifiUnwindingGatewayAdapter.sol	**Low**	Fixed
Unwinding cannot start simultaneously for several epochs	helpers/infinifi/InfinifiUnwindingGateway.sol	**Low**	Acknowledged
Wrong comment	helpers/uniswap/UniswapV4Gateway.sol	**Info**	Fixed

4. General Recommendations

This section contains general recommendations on how to improve 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. setLockedTokenBatchStatus doesn't remove tokens from unwindingEpochToLockedToken

Risk Level: Low

Status: Fixed in the commit.

Contracts:

- adapters/infinifi/InfinifiGatewayAdapter.sol
- adapters/infinifi/InfinifiUnwindingGatewayAdapter.sol

Location:

- `setLockedTokenBatchStatus`

Description:

The `setLockedTokenBatchStatus()` function doesn't remove the locked tokens from `unwindingEpochToLockedToken` mapping when a token is being disabled:

```
 if (lockedTokens[i].allowed) {
 _allowedLockedTokens.add(lockedTokens[i].lockedToken);
 _getMaskOrRevert(lockedTokens[i].lockedToken);
 unwindingEpochToLockedToken[lockedTokens[i].unwindingEpochs] =
lockedTokens[i].lockedToken;
 lockedTokenToUnwindingEpoch[lockedTokens[i].lockedToken] =
lockedTokens[i].unwindingEpochs;
 } else {
 _allowedLockedTokens.remove(lockedTokens[i].lockedToken);
 delete lockedTokenToUnwindingEpoch[lockedTokens[i].lockedToken];
 }
```

As a result, `unwindingEpochToLockedToken` may retain a stale reference after the token is removed from the allowed set.

Remediation:

Consider removing the entry from `unwindingEpochToLockedToken`:

```
} else {
 _allowedLockedTokens.remove(lockedTokens[i].lockedToken);
 delete lockedTokenToUnwindingEpoch[lockedTokens[i].lockedToken];
++ delete
unwindingEpochToLockedToken[lockedTokens[i].unwindingEpochs];
 }
```

5.2. Unwinding cannot start simultaneously for several epochs

Risk Level: Low

Status: This is intentional, to avoid high gas costs when calculating the balance of the phantom token.

Contracts:

- helpers/infinifi/InfinifiUnwindingGateway.sol

Location:

- `startUnwinding`

Description:

The `startUnwinding()` function of InfinifiUnwindingGateway verifies on line 50 that `msg.sender` does not already have an active unwinding process.

```
contracts/helpers/infinifi/InfinifiUnwindingGateway.sol:
  46: function startUnwinding(uint256 shares, uint32 unwindingEpochs)
external {
  47: if (block.timestamp == lastUnwindingTimestamp) revert
MoreThanOneUnwindingPerBlockException();
  48:
  49: UserUnwindingData storage userUnwindingData =
userToUnwindingData[msg.sender];
  50: if (userUnwindingData.unwindingTimestamp != 0) revert
UserAlreadyUnwindingException();
  51:
  52: address lockedToken =
IInfinifiLockingController(lockingController).shareToken(unwindingEpochs);
  53:
  54: IERC20(lockedToken).transferFrom(msg.sender, address(this), shares);
  55: IERC20(lockedToken).approve(infinifiGateway, shares);
  56:
  57: userUnwindingData.shares = shares;
  58: userUnwindingData.unwindingTimestamp = block.timestamp;
  59: userUnwindingData.isWithdrawn = false;
  60: userUnwindingData.unwindingEpochs = unwindingEpochs;
  61: lastUnwindingTimestamp = block.timestamp;
  62:
  63: IInfinifiGateway(infinifiGateway).startUnwinding(shares,
unwindingEpochs);
  64: }
```

This check prevents users from having more than one pending unwinding of a locked position token. If a user has created multiple positions and wants to unlock them all at once, the function will not allow it. Instead, each position must be unlocked sequentially, which could take several weeks to complete.

Remediation:

Consider refactoring the logic to allow users to unlock multiple positions in parallel.

5.3. Wrong comment

Risk Level: Info

Status: Fixed in the commit.

Contracts:

- helpers/uniswap/UniswapV4Gateway.sol

Description:

The comment is incorrectly states that the contract is a connector for the Balancer V3 Router:

```
contracts/helpers/uniswap/UniswapV4Gateway.sol:
  27: /// @dev This is connector contract to allow Gearbox adapters to
swap through the Balancer V3 Router.
```

Remediation:

Consider updating the comment to correctly reflect the purpose of the contract.

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.