Sky: LZ Governance EVM

Security Review

Cantina Managed review by:

Christoph Michel, Lead Security Researcher
Mario.eth, Lead Security Researcher

October 28, 2025

Contents

1 Introduction **2**
1.1 About Cantina . . . 2
1.2 Disclaimer . . . 2
1.3 Risk assessment . . . 2
1.3.1 Severity Classification . . . 2

2 Security Review Summary **3**

3 Findings **4**
3.1 Informational . . . 4
3.1.1 Unchecked `target` on destination chain could be used to manage `GovernanceOAppReceiver`'s LZ configuration . . . 4

1 Introduction

1.1 About Cantina

Cantina is a security services marketplace that connects top security researchers and solutions with clients. Learn more at cantina.xyz

1.2 Disclaimer

Cantina Managed provides a detailed evaluation of the security posture of the code at a particular moment based on the information available at the time of the review. While Cantina Managed endeavors to identify and disclose all potential security issues, it cannot guarantee that every vulnerability will be detected or that the code will be entirely secure against all possible attacks. The assessment is conducted based on the specific commit and version of the code provided. Any subsequent modifications to the code may introduce new vulnerabilities that were absent during the initial review. Therefore, any changes made to the code require a new security review to ensure that the code remains secure. Please be advised that the Cantina Managed security review is not a replacement for continuous security measures such as penetration testing, vulnerability scanning, and regular code reviews.

1.3 Risk assessment

Severity level	Impact: High	Impact: Medium	Impact: Low
Likelihood: high	Critical	High	Medium
Likelihood: medium	High	Medium	Low
Likelihood: low	Medium	Low	Low

1.3.1 Severity Classification

The severity of security issues found during the security review is categorized based on the above table. Critical findings have a high likelihood of being exploited and must be addressed immediately. High findings are almost certain to occur, easy to perform, or not easy but highly incentivized thus must be fixed as soon as possible.

Medium findings are conditionally possible or incentivized but are still relatively likely to occur and should be addressed. Low findings are a rare combination of circumstances to exploit, or offer little to no incentive to exploit but are recommended to be addressed.

Lastly, some findings might represent objective improvements that should be addressed but do not impact the project's overall security (Gas and Informational findings).

2 Security Review Summary

Sky Protocol is a decentralised protocol developed around the USDS stablecoin.

From Sep 22nd to Sep 26th the Cantina team conducted a review of sky-oapp-oft on commit hash efc7a928. The team identified a total of **1** issues:

Issues Found

Severity	Count	Fixed	Acknowledged
Critical Risk	0	0	0
High Risk	0	0	0
Medium Risk	0	0	0
Low Risk	0	0	0
Gas Optimizations	0	0	0
Informational	1	0	1
Total	**1**	**0**	**1**

The Cantina Managed team reviewed Sky's `sky-oapp-oft` holistically on commit hash 03fa6e86 and concluded that all findings were addressed and no new vulnerabilities were identified.

3 Findings

3.1 Informational

3.1.1 Unchecked `target` on destination chain could be used to manage `GovernanceOAppReceiver`'s LZ configuration

Severity: Informational

Context: GovernanceOAppReceiver.sol#L90

Description: The `GovernanceOAppReceiver` can be used by multiple parties requiring cross-chain governance solutions by having different senders call `GovernanceOAppSender`. The `GovernanceOApp*` contracts are the paired "OApps" from LayerZero's point of view, therefore, the explicit delegate and the contract itself can manage its configurations (and other privileged actions) on the LZ endpoint:

```
// in LZ EndpointV2.sol
function _assertAuthorized(address _oapp) internal view override(MessagingChannel, MessageLibManager) {
    // oapp contract itself can also manage
    if (msg.sender != _oapp && msg.sender != delegates[_oapp]) revert Errors.LZ_Unauthorized();
}
```

> "For the `GovernanceOAppReceiver`, owner and delegate will be controlled by L2GovernanceRelay." Authority Setup. More precisely, the entire Governance OApp settings should be controlled by the specific `L2GovernanceRelay` "paired" to Sky Core's `L1GovernanceRelay` sender.

Note that on the destination chain, for `GovernanceOAppReceiver`, there is no check on the `dstTarget` contract that the OApp calls:

```
(bool success, bytes memory returnData) = dstTarget.call{ value: msg.value }(dstCallData);
```

Theoretically, `dstTarget` could be the `EndpointV2` contract and its `setConfig` (or similar actions requiring authorization) could be called to change the configuration, by a sender on the source chain that is *not* Sky Core.

In practice, the protection against unintended senders calling the `Endpoint` on the destination chain is set on the source chain only through the `canCallTarget[msg.sender][_params.dstEid][_params.dstTarget]` mapping in `GovernanceOappSender`.

Recommendation: Consider restricting the calls to any contract that is not the `EndpointV2` also on the destination chain in `GovernanceOAppReceiver._lzReceive`, such that the destination chain check is self-contained and does not rely on a correct configuration on the source chain.

Ensure `GovernanceOappSender.owner` is always managed by Sky Core itself, as otherwise it will lead to a privilege escalation (as the party has write access to the `canCallTarget` setting).

Sky: The intended purpose is that the `L2GovernanceRelay` will be used to manage everything (including the endpoint configs). However to maintain flexibility and because this can technically be guarded on `src`, that the tradeoff was worth it.

Cantina Managed: Acknowledged.