CANTINA

Bitlazer contracts feat/upgreadeable lbtc contract

Security Review

Cantina Managed review by:
Phaze, Security Researcher
Windhustler, Security Researcher

March 17, 2025

Contents

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	Description
Critical	*Must* fix as soon as possible (if already deployed).
High	Leads to a loss of a significant portion (>10%) of assets in the protocol, or significant harm to a majority of users.
Medium	Global losses <10% or losses to only a subset of users, but still unacceptable.
Low	Losses will be annoying but bearable. Applies to things like griefing attacks that can be easily repaired or even gas inefficiencies.
Gas Optimization	Suggestions around gas saving practices.
Informational	Suggestions around best practices or readability.

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 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

Bitlazer is a supercharged Bitcoin yield with Layer 3 speed and ultra-low transaction fees.

From Dec 21st to Dec 22nd the Cantina team conducted a review of bitlazer-contracts[feat/upgreadeable-lbtc-contract] on commit hash e61857e3. The team identified a total of **2** issues in the following risk categories:

- Critical Risk: 0
- High Risk: 0
- Medium Risk: 0
- Low Risk: 0
- Gas Optimizations: 0
- Informational: 2

3 Findings

3.1 Informational

3.1.1 Add storage gaps to `ZBTC` contract

Severity: Informational

Context: lBTC.sol#L96

Description: Storage gaps are a convention for reserving storage slots in a base contract, allowing future versions of that contract to use up to those slots without affecting the storage layout of child contracts.

With storage gaps, if someone inherits from `ZBTC` they can add new variables to the `ZBTC` contract without affecting its storage layout.

Recommendation: Add the storage gaps at the bottom of the contract:

```
      // Return WBTC to original holder
      WBTC.safeTransfer(msg.sender, amount);
  }
+
+ uint256[48] private __gap;
```

T3rn: Acknowledged.

Cantina Managed: Acknowledged.

3.1.2 ZBTC token can become insolvent due to unbacked minting

Severity: Informational

Context: *(No context files were provided by the reviewer)*

Description: The ZBTC system allows minting of new tokens through SZBTC conversion without ensuring sufficient WBTC backing exists in the contract. This creates a risk of system insolvency where the total ZBTC supply could exceed the WBTC reserves.

The ZBTC contract implements two ways to mint new tokens:

1. Standard minting that requires WBTC backing:

```
function mint(uint256 amount) public nonReentrant {
    require(!paused, "Contract paused");
    _mint(msg.sender, amount);
    WBTC.safeTransferFrom(msg.sender, address(this), amount);
}
```

2. Additional minting through SZBTC conversion:

```
function addExtraHolderBalance(uint256 amount) public nonReentrant {
    require(!pausedExtraHolderBalance, "Extra holder balance paused");
    uint256 sZBTCBalance = sZBTC.balanceOf(msg.sender);
    require(sZBTCBalance >= amount, "Insufficient sZBTC balance");
    // Burn the sZBTC
    sZBTC.burn(msg.sender, amount);
    // Mint the lBTC to the holder
    _mint(msg.sender, amount);
}
```

The issue is that SZBTC can be minted without restrictions by the owner:

```
function mint(address to, uint256 amount) public onlyOwner {
    _mint(to, amount);
}
```

There are no checks to ensure that enough WBTC exists in the contract to back newly minted ZBTC tokens created through SZBTC conversion. This means the total ZBTC supply could exceed the WBTC backing.

Recommendation: Consider implementing additional verification of sufficient WBTC backing before allowing any new ZBTC to be minted through SZBTC conversion.

T3rn: Acknowledged. The very fundamentals of szBTC is to act as the burnt supply of lzrBTC add-on, where on the side of Bitlazers Arbitrum Orbit network szBTC supply is correlated with the amount of lzrBTC burnt by the network users, and redistributed amongst all the lzrBTC stakars. Practically therefore, the total supply of lzrBTC and WBTC doesn't change, since the burnt for gas fees lzrBTC changed the ownership via redistribution to stakers. Stakers can bridge szBTC using Arbitrum Orbit L3 bridge back to Arbitrum Orbit and use it to increase their withdrawal limit of WBTC, but the total amount withdrawn of WBTC cannot exceed the original deposits, since the total szBTC supply cannot exceed supply burnt on gas fees

Cantina Managed: Acknowledged.