SOLIDIFIED

Audit Report for Aztec - May 11, 2022

Summary

Audit Report prepared by Solidified covering the Aztec protocol Ethereum Bridge contract for Set Bridge.

The following report covers the **Set Bridge**.

Process and Delivery

Three (3) independent Solidified experts performed an unbiased and isolated audit of the code. The debrief on 11 May 2022.

Audited Files

The source code has been supplied in the form of one public Github repository.

https://github.com/aztecProtocol/aztec-connect-bridges/

Commit Hash: `6c295e8e8eb4f1647de4ed8d3fdef7142fb555b7`

```
src
|-- bridges
|   -- set
       |-- IssuanceBridge.sol
       |-- interfaces
```

Intended Behavior

Smart contract responsible for depositing, managing and redeeming Defi interactions with the Set protocol.

Code Complexity and Test Coverage

Smart contract audits are an important step to improve the security of smart contracts and can find many issues. However, auditing complex codebases has its limits and a remaining risk is present (see disclaimer).

Users of a smart contract system should exercise caution. In order to help with the evaluation of the remaining risk, we provide a measure of the following key indicators: **code complexity**, **code readability**, **level of documentation**, and **test coverage**.

Note, that high complexity or lower test coverage does equate to a higher risk. Certain bugs are more easily detected in unit testing than in a security audit and vice versa. It is, therefore, more likely that undetected issues remain if the test coverage is low or non-existent.

Criteria	Status	Comment
Code complexity	Low	-
Code readability and clarity	High	-
Level of Documentation	Medium	-
Test Coverage	High	-

Issues Found

Issue #	Description	Severity	Status
1	The minimum acceptable amount of tokens to be received from Set's ExchangeIssuance exchange is specified as 0	Minor	Resolved
2	The actual amount of tokens received from ExchangeIssuance exchange is not checked	Minor	Acknowledged
3	Notes on code improvement	Note	

Critical Issues

No issues found

Major Issues

No issues found

Minor Issues

1. The minimum acceptable amount of tokens to be received from Set's ExchangeIssuance exchange is specified as 0

The function `convert()` specifies `0` as an acceptable minimum amount of tokens (or ETH) to be received from Set's `ExchangeIssuance` swaps.

Recommendation
Consider if it would be worth introducing a mechanism to specify a minimum amount of tokens/ETH received from the `ExchangeIssuance` swaps.

2. The actual amount of tokens received from ExchangeIssuance exchange is not checked

The function `convert()` does not check the amount of tokens received from the `ExchangeIssuance` exchange when doing the swaps.

Recommendation
Consider checking the amount of tokens received from the `ExchangeIssuance` exchange.

Informational Notes

3. Notes on code improvement

- `SafeMath` library import is not needed.
- function `convert` at line `#106` could use `else if` instead of `if`
- `revert` Error types would use less gas than `revert` messages (like in other bridges)
- Misleading comments like `// Check that spending of the given SetToken is approved` (actual approval is happening here)
- function `convert` - simplify complex if conditions with `require` or custom error types
- Consider custom `error` types instead of require messages with strings

Disclaimer

Solidified audit is not a security warranty, investment advice, or an endorsement of Aztec Protocol or its products. This audit does not provide a security or correctness guarantee of the audited smart contract. Securing smart contracts is a multistep process, therefore running a bug bounty program as a complement to this audit is strongly recommended.

The individual audit reports are anonymized and combined during a debrief process, in order to provide an unbiased delivery and protect the auditors from legal and financial liability.

Oak Security GmbH