SOLIDIFIED

Audit Report for EDuCoin - March 17, 2023

Summary

Audit Report prepared by Solidified covering the EDuCoin smart contract.

Process and Delivery

Two (2) independent Solidified experts performed an unbiased and isolated audit of the code below. The final debrief took place on March 17, 2023, and the results are presented here.

Audited Files

The source code has been supplied in a private source code repository:

https://github.com/opencampus-xyz/EDU-smart-contract/blob/main/contracts/EDuCoin.sol

Commit number: `6fd45c2db3a1e41113135151814bcbc9e618f1f0`

Intended Behavior

TinyTap EDuCoin is a fungible token that conforms to the ERC-20 standard and is based on the *Animoca Ethereum Contracts* library.

Findings

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 not necessarily equate to a higher risk, although certain bugs are more easily detected in unit testing than a security audit and vice versa.

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

Issues Found

Solidified found that the EDuCoin contracts contain no critical issues, no major issues, 0 minor issues, and 1 informational notes.

We recommend issues are amended, while informational notes are up to the team's discretion, as they refer to best practices.

Issue #	Description	Severity	Status
1	Redundant initializer calls in constructor()	Note	-

Critical Issues

No critical issues were found.

Major Issues

No major issues were found.

Minor Issues

No minor issues were found.

Informational Notes

1. Redundant initializer calls in `constructor()`

Initializers that take no arguments do not need to be explicitly called in the constructor and are hence redundant.

Recommendation
Consider removing redundant calls to the following initializers in `constructor()`: `ERC20()` and `ERC20Metadata()`.

Audit Report for EDuCoin - March 17, 2023

Disclaimer

Solidified audit is not a security warranty, investment advice, or an endorsement of TinyTap 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 of Solidified platform from legal and financial liability.

Oak Security GmbH