SOLIDIFIED

Audit Report for The Sandbox Claimable ERC-1155 Contract - December 10, 2020

Summary

Audit Report prepared by Solidified covering the Sandbox Claimable ERC-1155 contract.

Process and Delivery

Three (3) independent Solidified experts performed an unbiased and isolated audit of the code below. The debrief took place on December 7, 2020, and the results are presented here.

Audited Files

The following contracts were covered during the audit:

```
AssetGiveaway
├── AssetGiveaway.sol
└── ClaimERC1155.sol
```

The latest commit number covered by this report is:
`faa043859160e8fe7ae540dc853941da796de058`

Intended Behavior

The smart contract implements a wrapper to make ERC-1155 by submitting a Merkle proof.

SOLIDIFIED

Audit Report for The Sandbox Claimable ERC-1155 Contract - December 10, 2020

Executive Summary

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	Medium	-
Code readability and clarity	High	-
Level of Documentation	Medium	-
Test Coverage	High	-

SOLIDIFIED

Audit Report for The Sandbox Claimable ERC-1155 Contract - December 10, 2020

Issues Found

Solidified found that the Sandbox claimable ERC-1155 contract contains no security issues. We have identified 2 informational notes

Amendment of the notes are up to the team's discretion, as they refer to best practices or optimizations.

Issue #	Description	Severity	Status
1	Unused Import	Note	Resolved
2	Potential gas saving	Note	Resolved

Critical Issues

No critical issues have been found

Major Issues

No major issues have been found

Minor Issues

No major issues have been found

Notes

1. Unused Import

The `SafeMath` library is imported but not used in any arithmetic calculations.

Recommendation
Consider removing unused imports to clean up the code.

Update
Fixed

2. Potential gas saving

Audit Report for The Sandbox Claimable ERC-1155 Contract - December 10, 2020

The function `_sendAssets` checks for `_assetHolder` in every transfer to determine the from address. This can be further simplified to save gas.

Recommendation
Consider assigning the `_assetHolder` address in the `constructor` by checking the constructor parameter.

```
if (assetHolder == address(0) _assetHolder = address(this);
else _assetHolder = assetHolder;
```

Update
Fixed

Disclaimer

Solidified audit is not a security warranty, investment advice, or an endorsement of TSB GAMING LTD 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.

Solidified Technologies Inc.