Immutable Internal Contract Audit

Immutable ERC 20

Internal Auditors	Peter Robinson
Assessment Date	March 28, 2024
Final Report	March 28, 2024

Previous audits: The Immutable ERC 20 contracts have not previously been audited. However, they minimally change the Open Zeppelin contracts that have been extensively audited. In particular, the contracts in this audit extend the v4.9 contracts.

List of all Open Zeppelin contracts audits:
https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/audits
Audit report for Open Zeppelin contracts v4.9:
https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/audits/2023-05-v4.9.pdf

Contents

Description

Immutable supplies two ERC 20 contracts:

- ImmutableERC20.sol: This contract provides no functionality over Open Zeppelin's ERC 20 contract. It is unusable by itself as the deployed contract would have zero supply. It exists so that game studios that wish to create a custom ERC 20 contract know which provides a reliable starting point implementation that they can extend.
- ImmutableERC20FixedSupply.sol: This contract extends Open Zeppelin's ERC 20 contract, by creating a supply of tokens that is assigned to an owner address. Additionally, it provides the concept of ownership, which is used to link deployed contracts to Immutable Hub accounts.

Scope

Commit: b7adf0d702ea71ae43b65f904c1b18d7cdfbb4a2

Asset	Description
Smart Contracts	ERC20 contracts: • ImmutableERC20.sol • ImmutableERC20FixedSupplyNoBurn.sol
Threat model	None

Team's Greatest Concerns

- The team would like a general review. They have no specific concerns.

Basic Contract Analysis

The table below analyzes[1] the contracts.

Type	File	Lines	nSLOC	Complexity
Contract	ImmutableERC20	19	5	4
Contract	ImmutableERC20Fixed SupplyNoBurn	42	13	11

[1] Analysis produced using the Solidity Metrics VSCode extension.

Smart Contracts

ImmutableERC20

ImmutableERC20 extends Open Zeppelin's ERC20 contract. It offers the same functions, with the same authorisation checks.

Roles: None.

External or Public Functions[2] that modify state:

Function Name	Function Selector	Authorisation Check	Notes
approve	095ea7b3	There is no authorisation check. However, the actions of the functions only operate on allowances or balances related to msg.sender.	
decreaseAllowance	a457c2d7		
increaseAllowance	39509351		
transfer	a9059cbb		
transferFrom	23b872dd	Msg.sender must have an allowance authorized by the token's owner.	

External or Public Functions[3] that do not modify state:

Function Name	Function Selector	Notes
allowance	dd62ed3e	
balanceOf	70a08231	
decimals	313ce567	
name	06fdde03	
symbol	95d89b41	
totalSupply	18160ddd	

Upgradeable checks: This contract is not upgradeable..

Analysis of code logic: The contract correctly configures the symbol and name of the token.

[2] The list of functions was determined using forge inspect ./<path>/<contract>.sol:<contract> methods
[3] The list of functions was determined using forge inspect ./<path>/<contract>.sol:<contract> methods

ImmutableERC20FixedSupplyNoBurn

ImmutableERC20FixedSupplyNoBurn extends Open Zeppelin's ERC20 and Ownable contracts. They offer the same functions, with the same authorisation checks, with the exception of renounceOwnership.

Roles: Owner

External or Public Functions[4] that modify state:

Function Name	Function Selector	Authorisation Check	Notes
approve	095ea7b3	There is no authorisation check. However, the actions of the functions only operate on allowances or balances related to msg.sender.	
decreaseAllowance	a457c2d7		
increaseAllowance	39509351		
renounceOwnership	715018a6	None	Function always reverts.
transfer	a9059cbb	There is no authorisation check. However, the actions of the functions only operate on allowances or balances related to msg.sender.	
transferFrom	23b872dd	Msg.sender must have an allowance authorized by the token's owner.	
transferOwnership	f2fde38b	Only owner	

External or Public Functions[5] that do not modify state:

Function Name	Function Selector	Notes
allowance	dd62ed3e	
balanceOf	70a08231	
decimals	313ce567	
name	06fdde03	
owner	8da5cb5b	
symbol	95d89b41	
totalSupply	18160ddd	

Upgradeable checks: This contract is not upgradeable..

[4] The list of functions was determined using forge inspect ./<path>/<contract>.sol:<contract> methods
[5] The list of functions was determined using forge inspect ./<path>/<contract>.sol:<contract> methods

Analysis of code logic: The contract correctly creates a fixed supply of ERC-20 tokens, securely assigned to an owner upon deployment. The hub owner, token name and token symbol are also set.

Abuser Test Cases and Findings

Transfer Tokens to Incorrect Address

Classification: Informational

Description: An attacker could encourage game players that owned tokens to mistakenly transfer the tokens to an address not related to any contract or any private key. The tokens will be forever inaccessible. The attacker does not receive any benefit beyond the total circulating supply of tokens being reduced.

Action: For game players using Immutable's Passport Wallet, initially, there is no feature in the user interface to allow for the transfer of assets. When this feature is created, game players will be warned if they want to transfer assets to an address that is not a contract address and not an address with some IMX balance.

Status: Acknowledged by the team.