Immutable Internal Contract Audit

ImmutableX Bridge v4

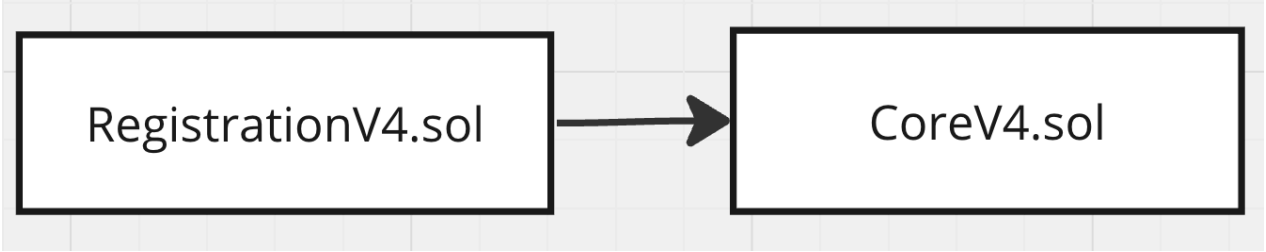
Internal Auditors	Peter Robinson
Assessment Date	February 16, 2024
Final Report	February 16, 2024

Previous audits: None.

Contents

Description

The Immutable-X v4 contracts provide functionality for enabling Immutable-X users to off-ramp assets from the StarkEx network to the Ethereum network. From v4 onwards, Starkex changed the deposit, withdrawal and registration flows to be more efficient and more trustless. The registration contract acts as a wrapper around the StarkEx contract to provide a more user-friendly interface. The architecture of the system is shown below.



Rather than users executing two Ethereum transactions, calling CoreV4 once for registration and then once for withdrawal, they can execute a single call RegistrationV4, which in the one transaction does the two calls to CoreV4.sol.

Scope

Commit: [8bb68f46ae9ed3add50ae8a6b1159f475bb063c5]

Asset	Description
Smart Contracts	https://github.com/immutable/contracts/tree/8bb68f46ae9ed3add50ae8a6b1159f475bb063c5/contracts/bridge/x/v4
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	RegistrationV4	70	43	48

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

Smart Contracts

RegistrationV4

RegistrationV4 does not extend any other contract.

Roles: None.

External or Public Functions[2] that modify state:

Function Name	Function Selector	Authorisation Check	Notes
registerAndWithdrawAll	022cabbc	Each function takes an Ethereum key, a Stark key, and a signature. If the Ethereum key has not yet been registered, the code verifies the signature, thus checking that the Stark key signed the Ethereum key, and registers the association between the two keys.. The withdrawal operation is then executed.	
registerAndWithdrawNft	352eb84c		
registerWithdrawAndMint	43fa186d		
withdrawAll	d2fc99b5	**None**	**See attacks section.**

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

Function Name	Function Selector	Notes
getVersion	0d8e6e2c	
imx	0f08025f	
isRegistered	579a6988	

Upgradeable checks: This contract is not upgradeable..

Analysis of code logic: Appears to be correct.

[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

Abuser Test Cases and Findings

Withdraw Authentication

Classification: Security

Description: The RegistrationV4.withdrawAll function checks the user's Eth and Stark balance, and then withdraws all of the user's Ether and Stark balance. This action is not authenticated. The only check is that for Stark withdrawals, the Stark library checks that there is an Ethereum key associated with the Stark key via the registration process.

An attacker could call this function passing in a user's Ethereum key, and have all of the user's Eth funds withdrawn. An attacker that knew that an Eth key had been registered for a Stark key could request a withdrawal by passing in the user's Stark key. In both cases, the user would unexpectedly have their funds withdrawn. The attacker could not steal the user's funds however.

Note: This is an issue that exists in Stark's library code. The RegistrationV4 wrapper code does not introduce this issue.

Action: Discuss with Stark team.

Status: Acknowledged by the team.

Stolen Stark Key and Unregistered Eth Key

Classification: Security

Description: Imagine the scenario: an attacker had stolen a user's Stark key, and the user has not yet registered their Ethereum key. The attacker can use the user's Stark key to sign their (that is the attacker's) Ethereum key, and call the Stark library to register their Ethereum key using the user's Stark key, the attacker's Eth key, and the signature. The attacker can then withdraw the user's Stark balance. Once one Eth key is registered against a Stark key, there is no method of changing the Eth key.

Action: Encourage user's to register their Eth key before they accumulate significant value.

Status: Acknowledged by the team.