MixBytes()

wstETH on Lisk Deployment Verification Report

APRIL 2025

Disclaimer

The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, investment advice, endorsement of the platform or its products, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bug free status. The audit documentation is for discussion purposes only. The information presented in this report is confidential and privileged. If you are reading this report, you agree to keep it confidential, not to copy, disclose or disseminate without the agreement of the Client. If you are not the intended recipient(s) of this document, please note that any disclosure, copying or dissemination of its content is strictly forbidden.

Scope

Network: Lisk

Scope:

Asset	Link	Comment
OssifiableProxy	0x9348AF23B01F2B517AFE8f29B3183d2Bb7d69Fcf	Ossifiable proxy for the L1ERC20TokenBridge
L1ERC20TokenBridge	0xC7315f4FaaB2F700fc6b4704BB801c46ff6327AC	Implementation of the L1ERC20TokenBridge
OssifiableProxy	0x76D8de471F54aAA87784119c60Df1bbFc852C415	Ossifiable proxy for the ERC20Bridged
ERC20Bridged	0x16B8006b49db9022BF5457BD2de0144a7d0F970b	WstETH
OssifiableProxy	0xca498Ee83eD3546321d4DC25e2789B0624F15f68	Ossifiable proxy for the L2ERC20TokenBridge
L2ERC20TokenBridge	0xE766BE7B76E3F4d06551CB169Dd69B10a58ba91D	Implementation of the L2ERC20TokenBridge
OptimismBridgeExecutor	0xfD050cDa025f6378e54ab5fd5Da377D242Ed74d3	Governance executor for the Lisk network

Audit reports:

OssifiableProxy, L1ERC20TokenBridge, ERC20Bridged, L2ERC20TokenBridge: report
OptimismBridgeExecutor: report

Deployment scripts:
OssifiableProxy, L1ERC20TokenBridge, ERC20Bridged, L2ERC20TokenBridge:
https://github.com/lidofinance/lido-l2/tree/a569a49966360fbd223f4bd26a8720eab3799e5f/scripts/lisk
OptimismBridgeExecutor:
https://github.com/lidofinance/governance-crosschain-bridges/blob/257ac7014f6742de99dd4fe840f97255bd61d6aa/deploy/gov-bridge-lisk.ts

Initialized roles:
`Proxy_admin` for `L1ERC20TokenBridge`: Lido Aragon Agent

`Proxy_admin` for `ERC20Bridged`, `L2ERC20TokenBridge`: OptimismBridgeExecutor

Lido Aragon Agent (0x3e40d73eb977dc6a537af587d48316fee66e9c8c) is granted `DEFAULT_ADMIN_ROLE`, `DEPOSITS_ENABLER_ROLE`, `DEPOSITS_DISABLER_ROLE`, `WITHDRAWALS_ENABLER_ROLE` and `WITHDRAWALS_DISABLER_ROLE` roles in the `L1ERC20TokenBridge` contract.

Emergency Brakes Multisig (0x73b047fe6337183a454c5217241d780a932777bd) is granted `DEPOSITS_DISABLER_ROLE` and `WITHDRAWALS_DISABLER_ROLE` in the `L1ERC20TokenBridge` contract.

Lido Multisig (0x1356C0b19c2531bBf0Dd23E585b7C7f7096EeC39) is granted `DEPOSITS_DISABLER_ROLE`, `WITHDRAWALS_DISABLER_ROLE` roles in the `L2ERC20TokenBridge` contract.

Optimism Bridge Executor (0xfD050cDa025f6378e54ab5fd5Da377D242Ed74d3) is granted `DEFAULT_ADMIN_ROLE`, `DEPOSITS_ENABLER_ROLE`, `DEPOSITS_DISABLER_ROLE`, `WITHDRAWALS_ENABLER_ROLE`, `WITHDRAWALS_DISABLER_ROLE` roles in the `L2ERC20TokenBridge` contract.

Verification checklist

☑ Network specific behavior

All the network features affecting the protocol's operation are being studied. The virtual machine, the message transmission process within the main network, and vice versa (all distinctive network features and how they can impact the protocol's operation) are being researched. A comparison of the network's operation is conducted for deployment with networks where the wstETH token has already been deployed.

Results

The Lisk Network is built on the OP stack with no significant differences from Optimism. The block time on Lisk is 2 seconds, which does not impact protocol security.

☑ Scope checking

This stage involves auditors researching the provided scope for verification, studying project dependencies, and building the protocol's architecture. Project documentation is examined. Existing tests are also run at this stage, and the test coverage level is checked. Contract mocks are investigated for logical errors. The protocol's architecture is examined for conceptual errors.

Results

The protocol architecture and documentation were previously examined during the deployment verification on the Mantle and Base networks. Since there are no differences in the scope, it can be considered secure. All tests and mocks have been implemented correctly.

☑ Audit report investigation

At this stage, the presence of an audit report is verified, along with the alignment of the scope in the report with the deployed scope. It is checked whether all critical vulnerabilities have either been fixed or there is evidence that the vulnerability cannot be fixed without posing a threat to the protocol. Recommendations and the conclusion in the report are studied, as well as the alignment of the final commit with all the recommendations.

Results

There are no unresolved issues in the presented audit report, and the audited version matches the deployed one. The report includes one Critical and one High-severity issue, both related to the Arbitrum Bridge, which is not used in the current implementation. The audit report for the `OptimismBridgeExecutor` includes issues of only Warning and Info severity levels.

☑ Deploy script check

Auditors study the deployment script for contracts, examining initialization parameters. It is verified that interrupting the protocol deployment will not lead to incorrect initialization (for example, a front-run on initialization should result in both the script's reversion and require re-deployment).

Results

The deployment script available at this link correctly deploys all the necessary contracts and initializes the proxy in a manner that prevents front-running attacks.

☑ Deployment verification

The bytecode of the deployed contracts is checked to match the final commit in the report. An additional check is performed to verify all contracts on the explorer. Further verification is conducted to confirm that the bytecode of deployed contracts cannot be altered (https://mixbytes.io/blog/metamorphic-smart-contracts-is-evm-code-truly-immutable).

Results

The bytecode of the deployed contracts fully matches the audited version. All contracts were deployed from an EOA, eliminating the risk of metamorphic contracts.

☑ Initialization parameters check

At this stage, values are gathered from the storage in verified contracts, and they are checked for compliance with the parameters from the deployment script. Auditors ensure that all contracts are initialized and cannot be reinitialized by malicious users.

Results

All contracts have been correctly initialized, and all implementations are protected against reinitialization. All parameters used to configure the contracts and set their initial values are accurate.

☑ Role model verification

The protocol's access control structure is examined to identify redundant roles or roles with more privileges than intended. It is checked that all access rights are set by the previously studied structure. If a role is assigned to a multisig, multisig owners are validated.

Results

All roles have been granted correctly to the appropriate addresses. There are no unknown addresses used in the configured multisig. The multisig owners configured on Lisk are the same as those used for Gnosis Safe multisig on the Ethereum mainnet.

Social Links

https://mixbytes.io/

https://github.com/mixbytes/audits_public

hello@mixbytes.io

https://x/mixbytes