Liberland Blockchain

Security Audit

Author: CCTF's Awalcon Security Team

Date: *2023. October 10.*

Table of Contents

Disclaimer....3

Audit overview....4

Executive summary....5

Objective and methodology....6

Risk Classifications....7

Audit results....8

- Critical severity....8
- High severity....8
- Medium severity....8
- Low severity....9
  - Update crates returned by Cargo audit results....9
- Informational and recommendations....10
  - [INFO] Changeability of logic flow – Runtime upgrades....10
  - [INFO] Potential Insufficiency of Running Nodes....11
  - [INFO] Utilization of Sudo....12
  - [INFO] Government admin and clerk roles have identity control....13

Contact....14

Disclaimer

The list of findings and recommendations are summarized in the Audit Results.

The matters raised in this report are only those identified during the review and are not necessarily a comprehensive statement of all weaknesses that exist or all actions that might be taken. This work was performed under limitations of time and scope that are not potentially relevant to the actions of a malicious attack.

The review is based at a specific point in time, in an environment where both the systems and the threat profiles are dynamically evolving. It is therefore possible that vulnerabilities exist or will arise that were not identified during the review and there may or will have been events, developments and changes in circumstances subsequent to its issue.

The security analysis is purely based on the provided git repository alone. No other products or systems have been reviewed. The purpose of the audit is to identify issues related to the logic and quality of the code.

Audit overview

Liberland which is the contracted project, requested a security code audit on their Substrate based blockchain implementation.

Start date of the audit: 2023.09.16.

Report date: 2023.10.10.

Project website: https://liberland.org/en/
Types: Rust / Substrate

Scope: https://github.com/liberland/liberland_substrate

Audited version: v14.0.0
https://github.com/liberland/liberland_substrate/releases/tag/v14.0.0

Auditors:

- six
- *G*
- *wigy*

Overall result: **Pass**

Executive summary

Security Audit for Liberland's Substrate Blockchain

The security audit was commissioned by Liberland to meticulously evaluate the security and integrity of their Substrate blockchain. This technological infrastructure endeavors to provide a secure and stable foundation for Liberland's government, digital assets and transactions, which is crucial for safeguarding citizen’s and official’s interests, ensuring secure operations across the platform.

Key Findings

- **No Critical or High-Level Vulnerabilities:** The audit did not unearth any critical or high-level vulnerabilities in the bridge’s implementation. This is pivotal as it indicates that the implemented bridge has a strong foundational structure and ensures a significant level of confidence in its operational integrity.
- **No Medium-Level Findings:** This audit explicitly omits any medium-level findings, ensuring a concentrated focus on elements that are of low severity to not distract from the significant assurances in the established security framework.

Implications

The absence of critical and high-level vulnerabilities is a testament to the strong architectural and coding practices employed during the development of the bridge. However, the low-level vulnerabilities found have the potential to cause unwanted events for the project and thus, prompt and considered actions were already taken by Liberland’s developer team. Addressing these weaknesses enhance the overall security of the bridge and reduce the risk of exploitation or malfunction in the future.

As addition, we have added our recommendations for the projects future success.

Objective and methodology

The objective of the security assessment is to gain insight into the security of the project listed in the scope.

Security review main check items:

- Line-by-line audit
- Manual testing
- Business logic
- Data consistency
- Coding style violations
- System crashes
- Test with automated tools:
  - Cargo audit
  - Cargo vet
  - Custom tools by Awalcon

Risk Classifications

Critical: Vulnerabilities that can lead to a loss of funds, impairment, or external control over the system or its function. We recommend that findings of this classification are fixed immediately.

High: Findings of this classification can impact the flow of logic and can cause direct disruption in the system and the project's organization. We recommend that issues of this classification are fixed as soon as possible.

Medium: Vulnerabilities of this class have impact on the flow of logic, but does not cause any disturbance that would halt the system or organizational continuity. We recommend that findings of this class are fixed nonetheless.

Low: Bugs, or vulnerability that have minimal impact and do not pose a significant threat to the project or its users. We recommend that issues of this class are fixed nonetheless because they increase the attack surface when your project is targeted by malicious actors.

Informational: Findings of this class have a negligible risk factor but refer to best practices in syntax, style or general security.

Audit results

Critical severity

No critical severity issue has been found during the manual code review or by using automated tools.

High severity

No high severity issue has been found during the manual code review or by using automated tools.

Medium severity

No medium severity issue has been found during the manual code review or by using automated tools.

Low severity

Update crates returned by Cargo audit results

Description

The security audit performed using Cargo audit has identified several crates that require updates, potentially due to known vulnerabilities, outdated features, or other issues.

While this is a low-severity finding, outdated crates could potentially expose the chain to known issues, even if they are not immediately exploitable or critical.

Crates related: ed25519-dalek, quinn-proto, time, webpki

Risks

We could not identify any viable exploit at this moment, but it is better to keep the security updates actual to lower the potential attack surface.

Also, continually neglecting updates will accumulate technical debt, which might complicate future update efforts.

Proposed solution

Update the project’s rust config with the latest crates.

Informational and recommendations

[INFO] Changeability of logic flow – Runtime upgrades

Description

In the blockchain implementation, the logic flow, for example the maximum supply of the coins and tokens cannot be “hardcoded” thanks to the feature of Substrate runtime upgrades. This is intentional choice by Substrate’s architecture, but to stay secure a strong community with the right incentivization is needed.

This was communicated and Liberland accepts the risks.

Risks

- In case the project wants to list the coins on a CEX, it might be seen as a drawback that the system can be changed by the community. Liberland will need a community large enough to stay safe with runtime upgrades, including a professional developers, similar to Polkadot’s Fellowship.

Recommendation

- Utilize the Collective pallet and utilize it to increase the security of the blockchain through access control.

- Take examples of already existing collectives, for example the Fellowship.

Reference

https://docs.substrate.io/maintain/runtime-upgrades/

https://forum.polkadot.network/t/calling-polkadot-core-developers/506

[INFO] Potential Insufficiency of Running Nodes

Description

It is observed that the Substrate blockchain network might be operating with a suboptimal number of nodes, which while not immediately critical, is important for maintaining network resilience and security. Although the live network is currently functioning, having a limited number of nodes can potentially compromise decentralization, reliability, and security.

Risks

Network Security: A low number of nodes decreases security of the chain against attacks (e.g., Sybil attacks).

Decentralization: Limited nodes can lead to a more centralized network.

Network Stability: Fewer nodes might impact network stability, security.

Recommendation

- Community Engagement: engage with the community and stakeholders to communicate the importance of running nodes, potentially let skilled teams run more nodes.

- Node Operator Incentivization: Implement or enhance incentives for running nodes to attract more participants.

- Diversification: Ensure nodes are geographically and organizationally distributed to safeguard decentralization.

Reference

https://polkadot.js.org/apps/?rpc=wss%3A%2F%2Frpc.polkadot.io#/staking

https://wiki.polkadot.network/docs/thousand-validators

[INFO] Utilization of Sudo

Description

The Sudo pallet in Substrate blockchains allows a single account (the "sudo" key) to execute dispatchable calls with `Root` origin. Although useful in a network's early stages or in testnets, using a Sudo pallet poses risks in a production environment. The current blockchain configuration utilizes the Sudo pallet, allowing a singular account notable permissions and capabilities, which might become a single point of failure or risk.

Risks

- Centralization: Sudo pallet usage could lead to centralized decision-making and control, which is often contrary to blockchain principles.

- Security Vulnerabilities: The Sudo account becomes a high-value target due to its elevated privileges, risking exploitation if compromised.

- Operational Risk: Mistakes or erroneous commands from the Sudo account could result in unintended network consequences.

- Reputation Damage: Perceptions of centralization or potential misuse of the Sudo pallet might harm the reputation of the blockchain among users and stakeholders.

- Legal and Compliance Risks: Centralized control might inadvertently place legal responsibilities on the entity controlling the Sudo key.

Proposed solution

- Implement multisig for the sudo account (already done by Liberland team)

- Replace the Sudo with a collective team of trusted and skilled individuals, only for those calls that are required by the blockchain.

[INFO] Government admin and clerk roles have identity control

Description

Similar issue just as explained above with sudo, the admin and clerk roles have strong power over the identities in the office. In case any of the gets breached or loses the keys, this might lead to expensive destruction on the blockchain. These roles can’t destroy the whole system and runtime upgrades can potentially solve potential incidents like that, but this would be very expensive and burdensome.

Risks

- Centralization of Power: Concentration of identity control within specific roles may contradict decentralization principles.

- Operational Risk: Mistakes or malicious actions by the admins/clerks could have cascading effects on user trust and system functionality.

- Legal and Compliance Risks: The mishandling of identity data might result in legal and compliance issues.

Recommendation

Role Segregation and Access Control:

- Principle of Least Privilege (PoLP): Ensure roles are assigned only the minimum levels of access - or permissions - needed to accomplish their tasks.

- Role Segregation: Differentiate roles and their privileges to ensure no single role has overarching control over identities.

- Multi-Signature Approvals: Implement multi-signature requirements for sensitive operations to avoid unilateral actions.

Contact

Awalcon Team - six

Website: *https://awalcon.org/*

E-mail: *six@cryptoctf.org*

Git: *https://git.hsbp.org/six*

***PGP**: B1F7 B1D6 8838 98B4 2212 1D90 CA71 D1E4 078E 99C5*

Awalcon Team - G

Website: *https://awalcon.org/*

E-mail: *g@cryptoctf.org*