FoChat - audit

Preliminary Comments

CertiK Assessed on May 21st, 2025

CertiK Assessed on May 21st, 2025

FoChat - audit

These preliminary comments were prepared by CertiK, the leader in Web3.0 security.

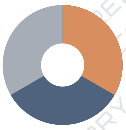
Executive Summary

TYPES	ECOSYSTEM	METHODS
Vault	Solana (SOL)	Formal Verification, Manual Review, Static Analysis

LANGUAGE	TIMELINE	KEY COMPONENTS
Rust	Delivered on 05/21/2025	N/A

CODEBASE	COMMITS
init	6ac05f7826f8f24a3f78bbea6be482bfb6d64b3e
View All in Codebase Page	View All in Codebase Page

Vulnerability Summary



3	0	0	0	0	3
Total Findings	Resolved	Partially Resolved	Acknowledged	Declined	Pending

1	Centralization	1 Pending	Centralization findings highlight privileged roles & functions and their capabilities, or instances where the project takes custody of users' assets.
0	Critical		Critical risks are those that impact the safe functioning of a platform and must be addressed before launch. Users should not invest in any project with outstanding critical risks.
0	Major		Major risks may include logical errors that, under specific circumstances, could result in fund losses or loss of project control.
0	Medium		Medium risks may not pose a direct risk to users' funds, but they can affect the overall functioning of a platform.
0	Minor		Minor risks can be any of the above, but on a smaller scale. They generally do not compromise the overall integrity of the project, but they may be less efficient than other solutions.
1	Informational	1 Pending	Informational errors are often recommendations to improve the style of the code or certain operations to fall within industry best practices. They usually do not affect the overall functioning of the code.
1	Discussion	1 Pending	The impact of the issue is yet to be determined, hence requires further clarifications from the project team.

TABLE OF CONTENTS | FOCHAT - AUDIT

CODEBASE | FOCHAT - AUDIT

Repository

init

Commit

6ac05f7826f8f24a3f78bbea6be482bfb6d64b3e

AUDIT SCOPE | FOCHAT - AUDIT

1 file audited ● 1 file with Pending findings

ID	Repo	File	SHA256 Checksum
● LIB	fodev2025/spl-lock	programs/lock/src/lib.rs	c64025056f21be639325ae025384462c14ae6acb49e54fb933cedc41566e6e02

APPROACH & METHODS | FOCHAT - AUDIT

This report has been prepared for FoChat to discover issues and vulnerabilities in the source code of the FoChat - audit project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Formal Verification, Manual Review, and Static Analysis techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Testing the smart contracts against both common and uncommon attack vectors;
- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

REVIEW NOTES | FOCHAT - AUDIT

System Overview

The `FoChat` program implements a token locking mechanism using the Anchor framework. Users can deposit tokens into a vault, locking them until a specified unlock time. It uses Program Derived Accounts (PDAs) to manage token authority and storage of lock metadata.

Out-of-Scope Components

The off-chain program, which responds to detect bridge-out events, sign the bridge-in packet, and distribute the signatures to the relayer, is not included in the audit scope. The audit team assumes the off-chain program is implemented securely.

External Dependencies

The project mainly contains the following dependencies:

Dependency	Version
anchor-lang	0.31.1
anchor-spl	0.31.1
solana-program	2.2.1

It should also be noted here that the code dependencies are in active development in the current auditing version and some of the keywords/functionality may be deprecated in a newer version. It is necessary to keep the dependencies up-to-date to avoid potential vulnerabilities.

The on-chain program can be upgradeable after the initial deployment due to Solana's features. Also, based on the unique rent mechanism in Solana, the balance in the account should be carefully set.

We assume these dependencies are valid and non-vulnerable factors and implement proper logic to collaborate with the current project. For example, the associated token account ownership transfer will not be considered after checking with the team.

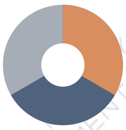
Privileged Functions

The **FoChat** project relies on upgrade authority to ensure the dynamic runtime updates of the project, which is specified in the **Centralization** finding.

To improve the trustworthiness of the project, the community should be notified of dynamic runtime updates. Any plans to invoke a privileged function should also be considered to move to the execution queue of a `Timelock` contract.

The Solana platform allows for upgrading its programs, with the default upgrade authority being the entity responsible for deployment. In situations where the program has upgradability features and the account of the upgrade authority becomes compromised, there is the potential for an unauthorized and malicious update to the program.

FINDINGS | FOCHAT - AUDIT



3	0	1	0	0	0	1	1
Total Findings	Critical	Centralization	Major	Medium	Minor	Informational	Discussion

This report has been prepared to discover issues and vulnerabilities for FoChat - audit. Through this audit, we have uncovered 3 issues ranging from different severity levels. Utilizing the techniques of Formal Verification, Manual Review & Static Analysis to complement rigorous manual code reviews, we discovered the following findings:

ID	Title	Category	Severity	Status
FOA-04	**Centralization Risk And Upgradeability**	**Centralization**	**Centralization**	**Pending**
FOA-03	Unused Error Code	Volatile Code	Informational	Pending
FOA-02	Discussion On User-Defined Unlock Time In Token Deposits	Logical Issue	Discussion	Pending

FOA-04 | CENTRALIZATION RISK AND UPGRADEABILITY

Category	Severity	Location	Status
Centralization	**Centralization**		**Pending**

Description

An Solana program can be deployed on the mainnet as:

- final: the code cannot be updated.
- upgradable: `BPFLoaderUpgradeable` is the program owner and an *upgrade authority*, a custom account, can upgrade the program code.

In case the `lock` program is deployed as upgradeable, the upgrade authority has the privilege to update the implementation of the programs at they will. Any compromise to the upgrade authority account may allow a hacker to take advantage of this authority and replace the implementation of the program and, therefore, execute any code on the program data and funds.

Recommendation

The risk describes the current project design and potentially makes iterations to improve in the security operation and level of decentralization, which in most cases cannot be resolved entirely at the present stage. We advise the client to carefully manage the privileged account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol be improved via a decentralized mechanism or program derived accounts with enhanced security practices, e.g., multi-signature wallets.

Indicatively, here are some feasible suggestions that would also mitigate the potential risk at a different level in terms of short-term, long-term and permanent:

Short Term:

Timelock and Multi sign (⅔, ⅗) combination *mitigate* by delaying the sensitive operation and avoiding a single point of key management failure.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key compromised;
  AND
- A medium/blog link for sharing the timelock contract and multi-signers addresses information with the public audience.

Long Term:

Timelock and DAO, the combination, *mitigate* by applying decentralization and transparency.

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
  AND
- Introduction of a DAO/governance/voting module to increase transparency and user involvement;
  AND
- A medium/blog link for sharing the timelock contract, multi-signers addresses, and DAO information with the public audience.

Permanent:

Renouncing the ownership or removing the function can be considered *fully resolved*.

- Renounce the ownership and never claim back the privileged roles;
  OR
- Remove the risky functionality.

Noted: Recommend considering the long-term solution or the permanent solution. The project team shall make a decision based on the current state of their project, timeline, and project resources.

FOA-03 | UNUSED ERROR CODE

Category	Severity	Location	Status
Volatile Code	● Informational	programs/lock/src/lib.rs: 265~266	● Pending

Description

The `BumpError` Error code is not used within the program.

Recommendation

It's recommended to either implement this error code for bump checks or remove it to improve code efficiency.

FOA-02 DISCUSSION ON USER-DEFINED UNLOCK TIME IN TOKEN DEPOSITS

Category	Severity	Location	Status
Logical Issue	Discussion	programs/lock/src/lib.rs: 12~13	Pending

Description

The `deposit()` function allows users to deposit any tokens into a vault account associated with a specific user and token mint. The input `unlock_time` provided by the user specifies the time when the tokens can be withdrawn from the vault. This input is validated by the following condition:

```
require!(
            unlock_time > clock.unix_timestamp,
            ErrorCode::UnlockTimeInPast
        );
```

This check ensures that the `unlock_time` is in the future, meaning it must be greater than the current Unix timestamp.

However, there are concerns about whether it is appropriate to allow users to determine their own unlock time for asset withdrawals, as the purpose of the lock/unlock behavior is unclear.

We would like to discuss this finding with the team to ensure that this design choice aligns with the intended business logic.

OPTIMIZATIONS | FOCHAT - AUDIT

ID	Title	Category	Severity	Status
FOA-01	Close Empty Vault Accounts To Save Costs	Design Issue	Optimization	Pending

FOA-01 | CLOSE EMPTY VAULT ACCOUNTS TO SAVE COSTS

Category	Severity	Location	Status
Design Issue	Optimization	programs/lock/src/lib.rs: 101~102	Pending

Description

According to the design logic, a `lock_account` is created for each specific user and token mint, serving as the authority and seed for the corresponding token vault. Each combination of user and token mint is associated with a unique `lock_account` and a unique `vault_token_account` . The `lock_account` stores the metadata for the vault token account, while the `vault_token_account` holds all the deposited tokens for the specific user and token.

When the user performs the `withdraw()` function, the data in the `lock_account` is cleared. However, the empty vault token account is not closed and remains on-chain.

Recommendation

To optimize on-chain costs, it is recommended to close the empty vault token account if it contains no tokens.

APPENDIX | FOCHAT - AUDIT

Finding Categories

Categories	Description
Volatile Code	Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases and may result in vulnerabilities.
Logical Issue	Logical Issue findings indicate general implementation issues related to the program logic.
Centralization	Centralization findings detail the design choices of designating privileged roles or other centralized controls over the code.
Design Issue	Design Issue findings indicate general issues at the design level beyond program logic that are not covered by other finding categories.

Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

DISCLAIMER | CERTIK

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK's position is that each company and individual are responsible for their own due diligence and continuous security. CertiK's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by CertiK is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE. WITHOUT LIMITATION TO THE FOREGOING, CERTIK PROVIDES NO WARRANTY OR

UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER CERTIK NOR ANY OF CERTIK'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS" AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.

THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT CERTIK'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.

FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

Elevating Your Entire Web3 Journey

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.