Cyberscope

Audit Report

Anso Finance

June 2025

Repository :
https://github.com/Quantum-Bases/solana-ico-contracts/tree/Audit-v1

Commit : e369b27e09286ab0e95d55be7584d126fd48648b

Audited by © cyberscope

Table of Contents

Risk Classification

The criticality of findings in Cyberscope's smart contract audits is determined by evaluating multiple variables. The two primary variables are:

1. **Likelihood of Exploitation**: This considers how easily an attack can be executed, including the economic feasibility for an attacker.
2. **Impact of Exploitation**: This assesses the potential consequences of an attack, particularly in terms of the loss of funds or disruption to the contract's functionality.

Based on these variables, findings are categorized into the following severity levels:

1. **Critical**: Indicates a vulnerability that is both highly likely to be exploited and can result in significant fund loss or severe disruption. Immediate action is required to address these issues.
2. **Medium**: Refers to vulnerabilities that are either less likely to be exploited or would have a moderate impact if exploited. These issues should be addressed in due course to ensure overall contract security.
3. **Minor**: Involves vulnerabilities that are unlikely to be exploited and would have a minor impact. These findings should still be considered for resolution to maintain best practices in security.
4. **Informative**: Points out potential improvements or informational notes that do not pose an immediate risk. Addressing these can enhance the overall quality and robustness of the contract.

Severity	Likelihood / Impact of Exploitation
Critical	Highly Likely / High Impact
Medium	Less Likely / High Impact or Highly Likely/ Lower Impact
Minor / Informative	Unlikely / Low to no Impact

Review

Repository	https://github.com/Quantum-Bases/solana-ico-contracts/tree/Audit-v1
Commit	e369b27e09286ab0e95d55be7584d126fd48648b

Audit Updates

Initial Audit	11 Jun 2025

Source Files

Filename	SHA256
lib.rs	61c20ed16751fe1d73019604f3247a98f2041cb1a87167c4e020bcd05cd26a95

Overview

The ANSO token sale contract implements a presale mechanism on Solana, designed to facilitate token distribution with a multi-stage pricing structure and vesting schedule. The protocol enables the presale of a token with predetermined pricing stages, and vesting rules, while enabling users to purchase tokens using various payment methods. The system includes features for payment token management, vesting schedules, and automated price adjustments through stage progression. This creates a controlled and structured way for the distribution of tokens while ensuring long-term alignment through vesting mechanisms.

Initialization

The protocol begins with the initialization of a `Sale` account that stores global parameters and settings. This initialization sets up crucial values such as payment token addresses, funding wallet, token prices, and cap parameters. The sale account also stores the owner's address and payment token configurations, establishing the administrative structure of the protocol.

Token Purchase Mechanism

Users can purchase tokens through the buy_tokens function, which implements a multi-stage pricing mechanism. The system calculates token amounts based on the current stage price, verifies payment token validity, and enforces purchase limits. The purchase process includes vesting schedule initialization, with 25% of tokens unlocked immediately after the purchase is concluded and the remaining 75% vested over three months. The system automatically progresses through pricing stages as tokens are sold, with each stage having a predetermined price increase.

Vesting System

The protocol implements a comprehensive vesting system where purchased tokens are subject to a time-based release schedule. The initial 25% of tokens are unlocked once the presale is concluded, while the remaining 75% are vested over three months. This mechanism ensures long-term holder alignment and prevents immediate token release. The system tracks claimed amounts and allows users to claim their vested tokens once they become available.

Payment Token Management

The protocol supports multiple payment tokens. Each payment token is configured with its own mint address, decimals, and active status. The system allows administrators to toggle payment token status and add new payment tokens up to a maximum of three.

Stage Progression

The protocol implements an automated stage progression system where token prices increase as more tokens are sold. Each stage has a predetermined price and token allocation. The system automatically advances to the next stage when the current stage's token allocation is exhausted, with prices increasing per stage.

Administrative Controls

The platform owner has significant administrative capabilities within the protocol. They can update sale status, modify payment token configurations, adjust cap parameters, and control the presale end status. The owner can also toggle payment token status and update soft and hard caps as needed.

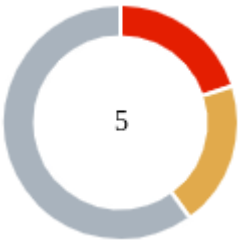
State Management

The protocol maintains several types of state accounts to track different aspects of the system. The `Sale` account stores global sale parameters and settings, while `VestingAccount` manages individual user vesting schedules. `UserPurchaseTracker` tracks user-specific purchase data, and various token accounts handle token balances.

Error Handling

The protocol implements comprehensive error handling through a custom error enum that covers various failure scenarios. These include payment token validation, arithmetic overflow protection, authorization checks, and business logic validation. The error system ensures robust operation and clear feedback for users and administrators.

Findings Breakdown



Critical	1
Medium	1
Minor / Informative	3

Severity	Unresolved	Acknowledged	Resolved	Other
Critical	1	0	0	0
Medium	0	1	0	0
Minor / Informative	3	0	0	0

Diagnostics

Critical Medium Minor / Informative

Severity	Code	Description	Status
Critical	USV	Uninitialized State Variable	Unresolved
Medium	MRO	Missing Refund Operation	Acknowledged
Minor / Informative	UPM	Uniform Pricing Mechanism	Unresolved
Minor / Informative	ICSP	Inconsistent Cross Stage Pricing	Unresolved
Minor / Informative	TSI	Tokens Sufficiency Insurance	Unresolved

USV - Uninitialized State Variable

Criticality	Critical
Location	lib.rs#L552
Status	Unresolved

Description

The contract defines the `total_tokens_for_sale` state variable within the `Sale` account, which is never initialized. As a result, its value remains undefined throughout execution. This oversight prevents the contract from accurately processing token sales, effectively rendering the presale mechanism non-functional.

```
let total_tokens_for_sale = sale.total_tokens_for_sale;Add commentMore actions
let tokens_sold = sale.tokens_sold_total;
require!(
tokens_sold.checked_add(amount).ok_or(CustomError::ArithmeticOverflow)? <=
total_tokens_for_sale,
CustomError::InsufficientSaleTokensAdd commentMore actions
);
```

Recommendation

The team should ensure that all state variables are properly initialized to maintain correct execution flow. Specifically, the `total_tokens_for_sale` variable should be initialized during the execution of the `initialize_sale` method.

MRO - Missing Refund Operation

Criticality	Medium
Location	lib.rs#L402
Status	Acknowledged

Description

The contract implements a sale mechanism with both a soft cap and a hard cap. However, it does not use these values to determine whether the presale has concluded successfully or failed to reach the hard cap. In cases where the hard cap is not met, the contract should facilitate refunds to users. Currently, tokens are non-refundable, and the presale can only be finalized by the owner regardless of the contributed amount. This may result in funds being locked indefinitely until the presale is concluded.

```
sale.is_presale_ended = true;
```

Recommendation

It is recommended to implement a mechanism that allows users to receive refunds if the hard cap is not reached within a reasonable timeframe. This would help build user trust in the sale mechanism.

Team Update

The team has acknowledged that this is not a security issue and states:

The refund operation is well documented and done manually by company multisig.

UPM - Uniform Pricing Mechanism

Criticality	Minor / Informative
Location	lib.rs#L160
Status	Unresolved

Description

The contract facilitates the sale of tokens by supporting multiple whitelisted payment tokens. It calculates the `total_cost` to be paid based on a uniform `token_price`. For consistent operation, this implies that all units of whitelisted tokens must hold the same monetary value. If a token with lesser underlying value is used, the consistency of the contract would be disrupted.

```
let numerator = amount_u128 * price_u128 *
10_u128.pow(payment_decimals);
let denominator = 10_u128.pow(token_decimals);
let result = numerator / denominator;
```

Recommendation

The team must ensure the consistency of operations by making sure all whitelisted tokens maintain the same underlying value.

ICSP - Inconsistent Cross Stage Pricing

Criticality	Minor / Informative
Location	lib.rs#L152
Status	Unresolved

Description

Users can purchase tokens through the `buy_tokens` method. When a certain limit of purchased tokens is exceeded, the contract advances to the next presale stage and increases the token price. However, if a user's purchase causes the current stage's limit to be exceeded, tokens from the next stage are still purchased at the current (lower) stage price. This behavior can lead to inconsistencies in the presale process and cause the contract's internal state to deviate.

```
let total_cost = {
let amount_u128 = amount as u128;
let price_u128 = sale.token_price as u128;
let token_decimals = sale.token_decimals as u32;
let payment_decimals = payment_token_decimals as u32;
...
}
```

Recommendation

The team is advised to ensure that sold tokens are consistently purchased at the expected price ratio. This can be achieved by modifying the current implementation to track the portion of tokens purchased during each stage.

TSI - Tokens Sufficiency Insurance

Criticality	Minor / Informative
Location	lib.rs#L609
Status	Unresolved

Description

The tokens are not held within the contract itself. Instead, the tokens are to be provided to the contract from an external administrator. While external administration can provide flexibility, it introduces a dependency on the administrator's actions, which can lead to various issues and centralization risks.

```
#[account(
init,
payer = owner,
associated_token::mint = anso_mint,
associated_token::authority = sale
)]
pub sale_anso_account: Account<'info, TokenAccount>,
```

Recommendation

It is recommended to consider implementing a more decentralized and automated approach for handling the contract tokens. One possible solution is to hold the tokens within the contract itself. If the contract guarantees the process it can enhance its reliability, security, and participant trust, ultimately leading to a more successful and efficient process.

Summary

The ANSO token sale contract is a token distribution protocol designed to manage the sale and release of tokens. It incorporates features such as time-based vesting schedules, a multi-stage pricing mechanism, support for multiple payment tokens, automated stage progression, purchase limits, and comprehensive error handling. This audit investigates security issues, business logic concerns and potential improvements.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. 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 Cyberscope’s prior written consent. 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 regarded as an indication of the economics or value of any “product” or “asset” created by any team or project that contracts Cyberscope to perform a security assessment. This document 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 represents an extensive assessment 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 Cyberscope’s position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope’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 Cyberscope are subject to dependencies and are 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.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

cyberscope.io