AUDIT REPORT

Vendetta Markets

Fixed Odds Market

Prepared by SCV-Security

On 30th January 2025

Table of Contents

Introduction

SCV has been engaged by Vendetta Markets to conduct a comprehensive security review with the goal of identifying potential security threats and vulnerabilities within the codebase. The purpose of this audit is to evaluate the security posture of the codebase and provide actionable recommendations to mitigate any identified risks. This report presents an overview of the findings from our security audit, outlining areas of concern and proposing effective measures to enhance the codebase's security.

Scope Functionality

This contract implements a fixed-odds betting market on the Cosmos blockchain using CosmWasm. It allows users to place bets on the outcomes of events (sports matches) with predefined odds, calculated based on the total liquidity and bets placed. The contract supports functionalities such as placing bets, claiming winnings, updating market parameters, scoring the event to determine the result, and canceling the market. It enforces rules like maximum bet limits, fee spreads, and initial odds, while ensuring only the admin can initialize or update the market. The contract also tracks bets, potential payouts, and market status, providing query functions to retrieve config, market details, and bet information.

Submitted Codebase

fixed Odds Market	
Repository	https://github.com/vendetta-labs/vendetta-markets-contracts
Contract	fixed-odds-market
Commit	`3730f741197307572ef75dd10a2697e8a1a11e31`
Branch	`main`

Revisions Codebase

fixed Odds Market	
Repository	https://github.com/vendetta-labs/vendetta-markets-contracts
Contract	fixed-odds-market
Commit	`d3d49beea151de019bd544ed9c9f48f761087a65`
Branch	`main`

Methodologies

SCV performs a combination of automated and manual security testing based on the scope of testing. The testing performed is based on the extensive experience and knowledge of the auditor to provide the greatest coverage and value to Vendetta Markets. Testing includes, but is not limited to, the following:

- Understanding the application and its functionality purpose.
- Deploying SCV in-house tooling to automate dependency analysis and static code review.
- Analyse each line of the code base and inspect application security perimeter.
- Review underlying infrastructure technologies and supply chain security posture.

Code Criteria

This section provides an evaluation of specific criteria aspects as described below:

- **Documentation:** Evaluating the presence and comprehensiveness of publicly available or provided explanatory information, diagram flowcharts, comments, and supporting documents to enhance code understanding.
- **Coverage:** Evaluating whether the code adequately addresses all necessary cases and scenarios, ensuring that the intended functionality or requirements are sufficiently covered.
- **Readability:** Assessing how easily the code can be understood and maintained, considering factors such as code structure, naming conventions, and overall organisation.
- **Complexity:** Evaluating the complexity of the code, including factors such as, number of lines, conditional statements, and nested structures.

The status of each criteria is categorised as either **SUFFICIENT** or **NOT-SUFFICIENT** based on the audit assessment. This categorisation provides insights to identify areas that may require further attention and improvement.

Criteria	Status	Notes
Documentation	**SUFFICIENT**	The codebase included thorough documentation that covered its functionality and expected conditions for each entrypoint.
Coverage	**SUFFICIENT**	Testing coverage is considered sufficient. 98.79% coverage, from 488 out of 494 lines covered.
Readability	**SUFFICIENT**	The codebase had good readability overall and utilised many Rust and CosmWasm best practices.
Complexity	**SUFFICIENT**	N/A

Findings Summary

Summary Title	Risk Impact	Status
Remove Hard-Coded Admin Address	MODERATE	RESOLVED
Remove Hard-Coded Treasury Address	LOW	RESOLVED
Additional Config Validations	LOW	RESOLVED

Findings Technical Details

1. Remove Hard-Coded Admin Address

RISK IMPACT: MODERATE	STATUS: RESOLVED

Description

In `contracts/fixed-odds-market/src/contract.rs:25` the contract admin address is hard-coded. This admin address is the privileged address that can both instantiate the fixed odds markets and perform admin functionality within the contract when it's live. This address is required at the time of the instantiation and is enforced so that no other address can instantiate a contract from the fixed odds market code-id, but this method of hard-coding the address can be improved to allow for more flexibility and operational security. In the event that the admin is compromised or needs to be changed in the future, hard-coding this address adds unnecessary complexity to the process.

Recommendation

Cosmwasm allows for an optional instantiate permission to be passed when a code-id is being stored. It is best practice to provide the privileged admin address during the store-code operation rather than hardcoding it in the contract.

2. Remove Hard-Coded Treasury Address

RISK IMPACT: LOW	STATUS: RESOLVED

Description

In `contracts/fixed-odds-market/src/contract.rs:26` the treasury address is hard-coded. The treasury address receives the market outstanding balances when a score is executed, and there is currently no way to update this address. This negatively impacts the manageability of the contract in the future.

Recommendation

We recommend adding the treasury address to the instantiation of the contract to ensure that the contract allows future updates if the treasury address were to need to change rather than requiring a migration or a new wasm code-id.

3. Additional Config Validations

RISK IMPACT: LOW	STATUS: RESOLVED

Description

The fixed odds market config does not have sufficient validations to ensure that misconfigurations are not introduced during the instantiation or update process.

`fee_spread_odds` should be validated to ensure it properly represents the fee spread in percentage points, and should not exceed a maximum value.

The initial odds (without the fee added) should be validated to ensure that their sum is one.

`seed_liquidity_amplifier` and `max_bet_risk_factor` should also be validated to ensure their initial values are within the range of expected values.

Recommendation

We recommend adding additional validation to the config parameters to ensure that the values supplied in the instantiation are within the range of expected values.

Document Control

Version	Date	Notes
-	20th January 2025	Security audit commencement date.
0.1	27th January 2025	Initial report with identified findings delivered.
0.5	30th January 2025	Fixes remediations implemented and reviewed.
1.0	30th January 2025	Audit completed, final report delivered.

Appendices

A. Appendix – Risk assessment methodology

SCV-Security employs a risk assessment methodology to evaluate vulnerabilities and identified issues. This approach involves the analysis of both the LIKELIHOOD of a security incident occurring and the potential IMPACT if such an incident were to happen. For each vulnerability, SCV-Security calculates a risk level on a scale of 5 to 1, where 5 denotes the highest likelihood or impact. Consequently, an overall risk level is derived from combining these two factors, resulting in a value from 10 to 1, with 10 signifying the most elevated level of security risk

Risk Level	Range
CRITICAL	10
SEVERE	From 9 to 8
MODERATE	From 7 to 6
LOW	From 5 to 4
INFORMATIONAL	From 3 to 1

LIKELIHOOD and **IMPACT** would be individually assessed based on the below:

Rate	LIKELIHOOD	IMPACT
5	**Extremely Likely**	Could result in severe and irreparable consequences.
4	**Likely**	May lead to substantial impact or loss.
3	**Possible**	Could cause partial impact or loss on a wide scale.
2	**Unlikely**	Might cause temporary disruptions or losses.
1	**Rare**	Could have minimal or negligible impact.

B. Appendix – Report Disclaimer

This report should not be regarded as an "endorsement" or "disapproval" of any specific project or team. These reports do not indicate the economics or value of any "product" or "asset" created by a team or project that engages SCV-Security for a security review. The audit report does not make any statements or warranties about the code's utility, safety, suitability of the business model, regulatory compliance of the business model, or any other claims regarding the fitness of the implementation for its purpose or its bug-free status. The audit documentation is intended for discussion purposes only. The content of this audit report is provided "as is," without representations and warranties of any kind, and SCV-Security disclaims any liability for damages arising from or in connection with this audit report. Copyright of this report remains with SCV-Security.

THANK YOU FOR CHOOSING

SCV SECURITY

scv.services

contact@scv.services