vacuumlabs

Cardano Casino

Audit Report v1.1

July 16, 2024

Contents

Revision table

Report version	Report name	Date	Report URL
1.1	Change of multiplier to a rational number	2024-07-16	Full report link
1.0	Main audit	2024-05-31	Full report link

1 Executive summary

THIS REPORT DOES NOT PROVIDE ANY WARRANTY OF QUALITY OR SECURITY OF THE AUDITED CODE and should be understood as a best efforts opinion of Vacuumlabs produced upon reviewing the materials provided to Vacuumlabs. Vacuumlabs can only comment on the issues it discovers and Vacuumlabs does not guarantee discovering all the relevant issues. Vacuumlabs also disclaims all warranties or guarantees in relation to the report to the maximum extent permitted by the applicable law. This report is also subject to the full disclaimer in the appendix of this document, which you should read before reading the report.

Project overview

There are no material changes in the project. Refer to the comprehensive project overview in the previous audit report. The most important change is the change of the `multiplier` type which was changed from an integer to a rational number. That allows for a more granular game setup.

Audit overview

The scope of this audit was limited to the changes in the commit `4c7a54a9cd`. The revision history for this report can be found in the list of revisions. Note that any findings from the previous revisions are not mentioned in this report.

The changes in this commit were very small. We verified that no new vulnerabilities were introduced by this change. Further, the Aiken compiler version was updated to the newest `v1.0.29-alpha+16fb02e` and the `stdlib` version to `1.9.0`. Even though the `stdlib` version lists breaking changes, we verified that it does not impact this project.

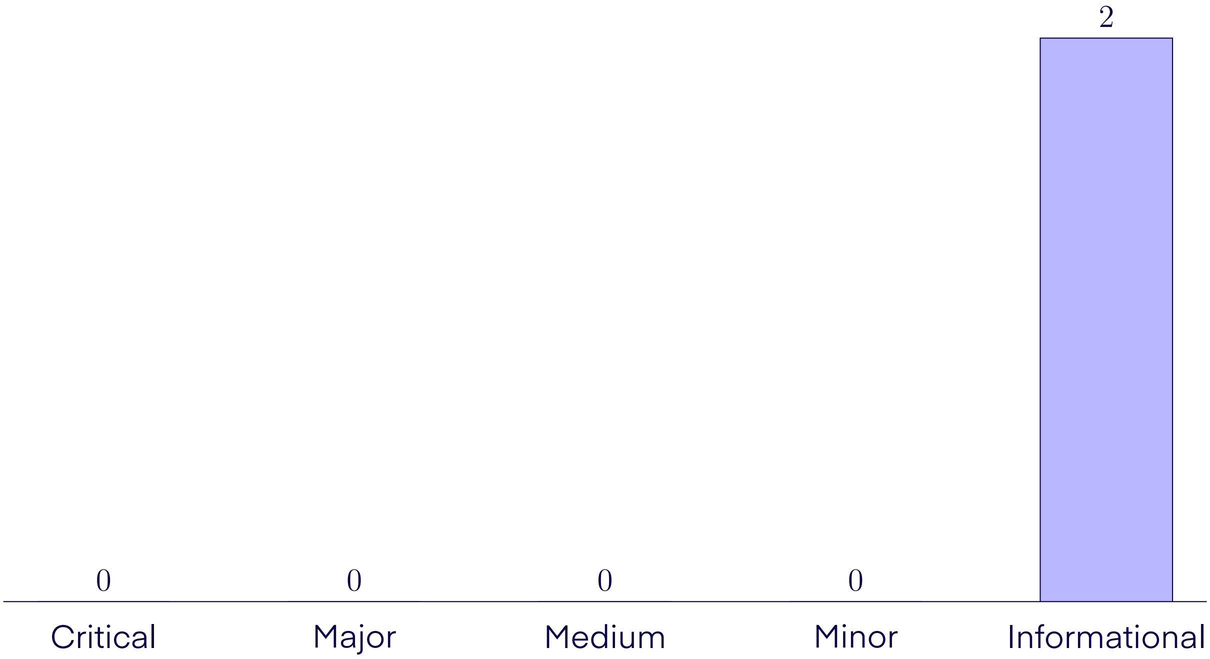
We started the audit at the previously audited commit, the commit `777d600408`, and it lasted from 15 July 2024 to 16 July 2024. We interacted mostly on Slack. The team fixed all issues to our satisfaction.

The commit `2547bca702ba2ba82626064694ebe0871c1c4015` represents the final version of the code. The status of any issue in this report reflects its status at that commit. You can see all the files audited and their hashes in Audited files. The smart contract language used is Aiken and the contracts are intended to run on Cardano. To avoid any doubt, we did not audit Aiken itself.

Summary of findings

During the audit, we found and reported 2 informational findings. All findings were fully resolved. We did not find any severe issues and only reported small improvements that do not pose security risks.

2 Severity overview



Findings

ID	TITLE	SEVERITY	STATUS
CCA2-401	Artefacts do not contain script version	INFORMATIONAL	RESOLVED
CCA2-402	Not formatted code	INFORMATIONAL	RESOLVED

CCA2-401 Artefacts do not contain script version

Category	Vulnerable commit	Severity	Status
Code Style	`4c7a54a9cd`	INFORMATIONAL	RESOLVED

Description

The version of the scripts had been updated in the `aiken.toml` version. However, the build process has not been run since, resulting in the `plutus.json` artefacts file not containing the v1.0.1 version tag.

Recommendation

We recommend recompiling the project with aiken build which bakes the 1.0.1 version specified in the `aiken.toml` file into the `plutus.json` file.

Resolution

The issue fix was present at the following commit: `2547bca702` .

CCA2-402 Not formatted code

Category	Vulnerable commit	Severity	Status
Code Style	`4c7a54a9cd`	INFORMATIONAL	RESOLVED

Description

The code changes have not been formatted using the aiken formatter: `aiken fmt`.

Recommendation

We recommend running the `aiken fmt` command that formats the code to a unified easier-readable style.

Resolution

The issue fix was present at the following commit: `2547bca702` .

A Disclaimer

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 agreement between VacuumLabs Bohemia s.r.o. (Vacuumlabs) and Cardano Casino (Client) (the Agreement), or the scope of services, and terms and conditions provided to the Client in connection with the Agreement, and shall be used only subject to and to the extent permitted by such terms and conditions. This report may not be transmitted, disclosed, referred to, modified by, or relied upon by any person for any purposes without Vacuumlabs's prior written consent.

This report is not, nor should be considered, an endorsement, approval or disapproval of any particular project, team, code, technology, asset or anything else. This report is not, nor should be considered, an indication of the economics or value of any technology, product or asset created by any team or project that contracts Vacuumlabs to perform a smart contract assessment. This report does not provide any warranty or guarantee regarding the quality or nature of the technology analysed, nor does it provide any indication of the technology's proprietors, business, business model or legal compliance.

To the fullest extent permitted by law, Vacuumlabs disclaims all warranties, expressed or implied, in connection with this report, its content, and the related services and products and your use thereof, including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. This report is provided on an as-is, where-is, and as-available basis. Vacuumlabs does not warrant, endorse, guarantee, or assume responsibility for any product or service advertised or offered by Client or any third party through the product, any open source or third-party software, code, libraries, materials, or information linked to, called by, referenced by or accessible through the report, its content, and the related services, assets and products, any hyperlinked websites, any websites or mobile applications appearing on any advertising, and Vacuumlabs will not be a party to or in any way be responsible for monitoring any transaction between you and client and/or any third-party providers of products or services.

This report should not be used in any way by anyone to make decisions around investment or involvement with any particular project, services or assets, especially not to make decisions to buy or sell any assets or products. This report provides general information and is not tailored to anyone's specific situation, its content, access, and/or usage thereof, including any associated services or materials, shall not be considered or relied upon as any form of financial, investment, tax, legal, regulatory, or other advice.

This report is based on the scope of materials and documentation provided for a limited review at the time provided. Vacuumlabs prepared this report as an informational exercise documenting the due diligence involved in the course of development of the Client's smart contract only, and This report makes no claims or guarantees concerning the smart contract's operation on deployment or post-deployment. This report provides no opinion or guarantee on the security of the code, smart contracts, project, the related assets or anything else at the time of deployment or post deployment. Smart contracts can be invoked by anyone on the internet and as such carry substantial risk. Vacuumlabs has no duty to monitor client's operation of the project and update the report accordingly.

The information contained in this report may not be complete nor inclusive of all vulnerabilities. This report is not comprehensive in scope, it excludes a number of components critical to the correct operation of this system. You agree that your access to and/or use of, including but not limited to, any associated services, products, protocols, platforms, content, assets, and materials will be at your sole risk. On its own, it cannot be considered a sufficient assessment of the correctness of the code or any technology. This report represents an extensive assessing process intending to help Client increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology, however blockchain technology and cryptographic assets present a high level of ongoing risk, including but not limited to unknown risks and flaws.

While Vacuumlabs has conducted an analysis to the best of its ability, it is Vacuumlabs's recommendation to commission several independent audits, a public bug bounty program, as well as continuous security auditing and monitoring and/or other auditing and monitoring in line with the industry best practice. The possibility of human error in the manual review process is highly real, and Vacuumlabs recommends seeking multiple independent opinions on any claims which impact any functioning of the code, project, smart contracts, systems, technology or involvement of any funds or assets. Vacuumlabs's position is that each company and individual are responsible for their own due diligence and continuous security.

B Audited files

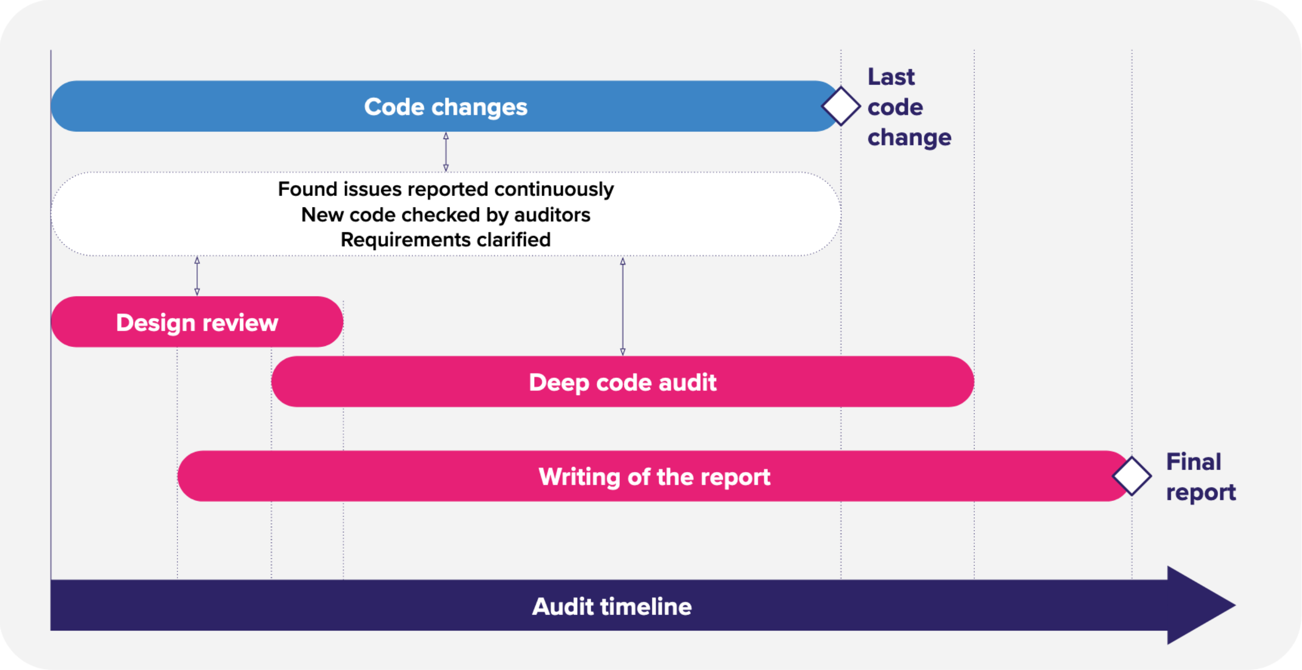
The files and their hashes reflect the final state at commit `2547bca702ba2ba82626064694ebe0871c1c4015` after all the fixes have been implemented.

SHA256 hash	Filename
30871...3f5af	lib/cardanocasino/types.ak
39adb...43fb2	validators/bets.ak

C Methodology

Vacuumlabs' agile methodology for performing security audits consists of several key phases:

1. Design reviews form the initial stage of our audits. The goal of the design review is to find larger issues which result in large changes to the code fast.
2. During the deep code audit, we verify the correctness of the given code and scrutinize it for potential vulnerabilities. We also verify the client's fixes for all discovered vulnerabilities. We provide our clients with status reports on a continuous basis providing them a clear up-to-date status of all the issues found so far.
3. We conclude the audit by handing over a final audit report which contains descriptions and resolutions for all the identified vulnerabilities.



Throughout our entire audit process, we report issues as soon as they are found and verified. We communicate with the client for the duration of the whole audit. During our audits, we check several key properties of the code:

- Vulnerabilities in the code
- Adherence of the code to the documented business logic
- Potential issues in the design that are not vulnerabilities
- Code quality

During our manual audits, we focus on several types of attacks, including but not limited to:

1. Double satisfaction
2. Theft of funds
3. Violation of business requirements
4. Token uniqueness attacks
5. Faking timestamps
6. Locking funds indefinitely
7. Denial of service
8. Unauthorized minting
9. Loss of staking rewards

D Issue classification

Severity levels

The following table explains the different severities.

Severity	Impact
CRITICAL	Theft of user funds, permanent freezing of funds, protocol insolvency, etc.
MAJOR	Theft of unclaimed yield, permanent freezing of unclaimed yield, temporary freezing of funds, etc.
MEDIUM	Smart contract unable to operate, partial theft of funds/yield, etc.
MINOR	Contract fails to deliver promised returns, but does not lose user funds.
INFORMATIONAL	Best practices, code style, readability, documentation, etc.

Resolution status

The following table explains the different resolution statuses.

Resolution status	Description
RESOLVED	Fix applied.
PARTIALLY RESOLVED	Fix applied partially.
ACKNOWLEDGED	Acknowledged by the project to be fixed later or out of scope.
PENDING	Still waiting for a fix or an official response.

Categories of issues

The following table explains the different categories of issues.

Category	Description
Design Issue	High-level issues in the design. Often large in scope, requiring changes to the design or massive code changes to fix.
Logical Issue	Medium-sized issues, often in between the design and the implementation. The changes required in the design should be small-scaled (e.g. clarifying details), but they can affect the code significantly.
Code Issue	Small in size, fixable solely through the implementation. This category covers all sorts of bugs, deviations from specification, etc.
Code Style	Parts of the code that work properly but are possible sources of later issues (e.g. inconsistent naming, dead code).
Documentation	Small issues that relate to any part of the documentation (design specification, code documentation, or other audited documents). This category does not cover faulty design.
Optimization	Ideas on how to increase performance or decrease costs.

E Report revisions

This appendix contains the changelog of this report. Please note that the versions of the reports used here do not correspond with the audited application versions.

v1.1: Change of multiplier to a rational number

Revision date: 2024-07-16
Final commit: `2547bca702ba2ba82626064694ebe0871c1c4015`

We conducted the audit of a small change: the `multiplier` type changed from an integer to a rational number. That allows for more granular game setup. To see the files audited, see Executive Summary.

Full report for this revision can be found at url.

v1.0: Main audit

Revision date: 2024-05-31
Final commit: `777d600408e8e87085d0ab506c167b4b71f28f26`

We conducted the audit of the main application.

SHA256 hash	Filename
54a93...d5c6a	lib/cardanocasino/types.ak
3289a...7d802	validators/bets.ak

Full report for this revision can be found at url.

F About us

Vacuumlabs has been building crypto projects since the early days.

- We helped create WingRiders, currently the second largest decentralized exchange on Cardano (based on TVL).
- We are behind the popular AdaLite wallet. It was later improved into a multichain wallet NuFi.
- We built the Cardano applications for the hardware wallets Ledger and Trezor.
- We built the first version of the cutting-edge decentralized NFT marketplace Jam On Bread on Cardano with truly unique features and superior speed of both the interface and transactions.

Our auditing team is chosen from the best.

- Talent from esteemed Cardano projects: WingRiders and NuFi.
- Rich experience across Google, traditional finance, trading and ethical hacking.
- Award-winning programmers from ACM ICPC, TopCoder and International Olympiad in Informatics.
- Driven by passion for program correctness, security, game theory and the blockchain technology.

vacuumlabs

Contact us:

audit@vacuumlabs.com