PUBLIC

Code Assessment

of the CDP Registry Smart Contracts

February 18, 2022

Produced for

by

Contents

1 Executive Summary

Dear Maker team,

Thank you for trusting us to help MakerDAO with this security audit. Our executive summary provides an overview of subjects covered in our audit of the latest reviewed contracts of CDP Registry according to Scope to support you in forming an opinion on their security risks.

MakerDAO implements a registry which allows users to only have one CDP ID per ilk. The registry will allow having an experience similar to the original proxy actions in the Charter and Cropper proxy actions.

The most critical subjects covered in our audit are functional correctness and access control. Security regarding all the aforementioned subjects is high.

The general subjects covered are gas efficiency, documentation and trustworthiness. Security regarding all the aforementioned subjects is high.

In summary, no issues were uncovered and we find that the codebase provides a high level of security.

It is important to note that security audits are time-boxed and cannot uncover all vulnerabilities. They complement but don't replace other vital measures to secure a project.

The following sections will give an overview of the system, our methodology, the issues uncovered and how they have been addressed. We are happy to receive questions and feedback to improve our service.

Sincerely yours,

ChainSecurity

1.1 Overview of the Findings

Below we provide a brief numerical overview of the findings and how they have been addressed.

Critical-Severity Findings	0
High-Severity Findings	0
Medium-Severity Findings	0
Low-Severity Findings	0

2 Assessment Overview

In this section, we briefly describe the overall structure and scope of the engagement, including the code commit which is referenced throughout this report.

2.1 Scope

The assessment was performed on the source code files inside the CDP Registry repository based on the documentation files. The table below indicates the code versions relevant to this report and when they were received.

V	Date	Commit Hash	Note
1	09 February 2022	7f60334b7b3c2b2816243672de2c808dc1f14ecf	Initial Version

For the solidity smart contracts, the compiler version `0.6.12` was chosen.

The file in scope was CdpRegistry.sol.

2.2 System Overview

This system overview describes the initially received version (Version 1) of the contracts as defined in the Assessment Overview.

Furthermore, in the findings section, we have added a version icon to each of the findings to increase the readability of the report.

So far, users have interacted with the VAT directly or have used the CdpManager to manage their vaults. For some new special ilks, direct interaction with the VAT will be prohibited and, hence, operations with such ilks will require calling contracts such as Charter or Cropper to ensure the validity of the operations on the VAT. Since the CdpManager can only interact with the VAT directly, a new registry CdpRegistry is introduced for Charter and Cropper which signals ownership of a vault and makes the experience of the Charter and Cropper proxy actions similar to the original proxy actions.

CdpRegistry stores the owner of a CDP and the ilk of a CDP. Note, that the CdpRegistry restricts users from having more than one vault per ilk. Hence, also a reverse mapping from user and ilk to the CDP ID is stored.

It only has function `open()` which opens a new CDP through the CdpManager. The id in the registry will be identical to the one in the manager. Note, that the owner on the CdpManager will be the CdpRegistry but in the CdpRegistry the owner will be the user. Even though CdpManager deploys a UrnHandler on `open()`, it remains unused since the contracts using the registry (e.g. Charter and Cropper) use the UrnProxy contracts they deployed.

3 Limitations and use of report

Security assessments cannot uncover all existing vulnerabilities; even an assessment in which no vulnerabilities are found is not a guarantee of a secure system. However, code assessments enable the discovery of vulnerabilities that were overlooked during development and areas where additional security measures are necessary. In most cases, applications are either fully protected against a certain type of attack, or they are completely unprotected against it. Some of the issues may affect the entire application, while some lack protection only in certain areas. This is why we carry out a source code assessment aimed at determining all locations that need to be fixed. Within the customer-determined time frame, ChainSecurity has performed an assessment in order to discover as many vulnerabilities as possible.

The focus of our assessment was limited to the code parts defined in the engagement letter. We assessed whether the project follows the provided specifications. These assessments are based on the provided threat model and trust assumptions. We draw attention to the fact that due to inherent limitations in any software development process and software product, an inherent risk exists that even major failures or malfunctions can remain undetected. Further uncertainties exist in any software product or application used during the development, which itself cannot be free from any error or failures. These preconditions can have an impact on the system's code and/or functions and/or operation. We did not assess the underlying third-party infrastructure which adds further inherent risks as we rely on the correct execution of the included third-party technology stack itself. Report readers should also take into account that over the life cycle of any software, changes to the product itself or to the environment in which it is operated can have an impact leading to operational behaviors other than those initially determined in the business specification.

4 Terminology

For the purpose of this assessment, we adopt the following terminology. To classify the severity of our findings, we determine the likelihood and impact (according to the CVSS risk rating methodology).

- *Likelihood* represents the likelihood of a finding to be triggered or exploited in practice
- *Impact* specifies the technical and business-related consequences of a finding
- *Severity* is derived based on the likelihood and the impact

We categorize the findings into four distinct categories, depending on their severity. These severities are derived from the likelihood and the impact using the following table, following a standard risk assessment procedure.

Likelihood	Impact		
	High	Medium	Low
High	Critical	High	Medium
Medium	High	Medium	Low
Low	Medium	Low	Low

As seen in the table above, findings that have both a high likelihood and a high impact are classified as critical. Intuitively, such findings are likely to be triggered and cause significant disruption. Overall, the severity correlates with the associated risk. However, every finding's risk should always be closely checked, regardless of severity.

5 Findings

In this section, we describe our findings. The findings are split into these different categories:

Below we provide a numerical overview of the identified findings, split up by their severity.

Critical-Severity Findings	0
High-Severity Findings	0
Medium-Severity Findings	0
Low-Severity Findings	0