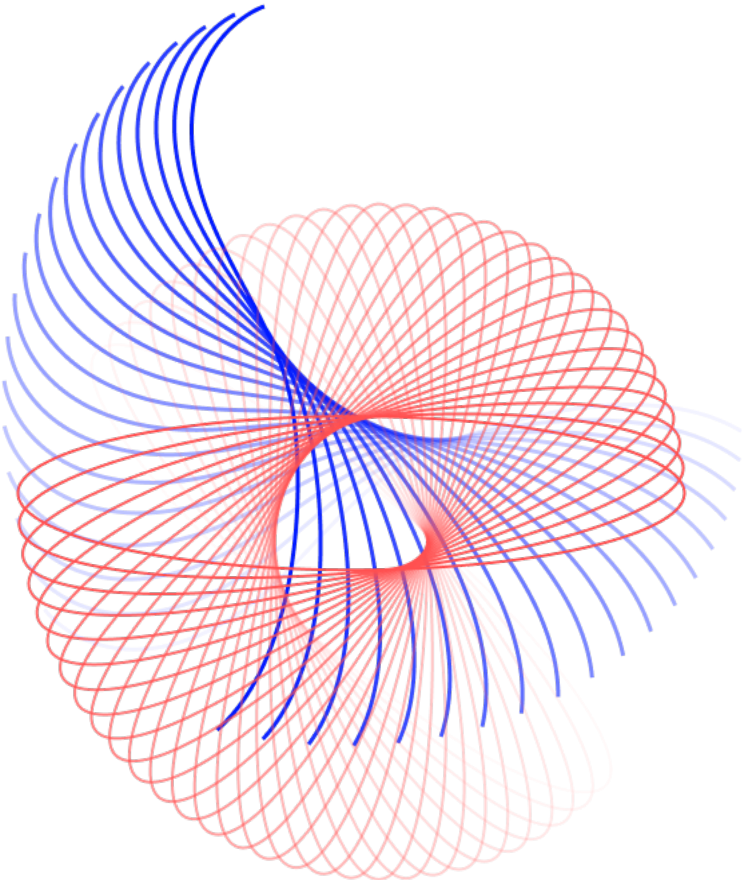
Security DAO

# CW3-Fixed-Multisig AUDIT

## CW3-Fixed Multisig in CosmWasm

Prepared by Logan Cerkownik

22 March 2022



secdao.xyz

# 1. Outline

## 1.1. Document revision history

| Version | Modification | Date | Author |
|---|---|---|---|
| 0.1 | Created | 3.22.2022 | Logan Cerkovnik |
| | | | |
| | | | |

## 1.2. Contact

| Contact | Organization | Email |
|---|---|---|
| Logan Cerkovnik | Security DAO | logan@secdao.xyz |
| Paul Wagner | Security DAO | paul@secdao.xyz |
| Barton Rhodes | Security DAO, DAO DAO | barton@secdao.xyz |

# Contents

# 2. Executive Overview

## 2.1. Audit Summary

Security Dao worked from 3/15/2022 through 3/23/2022 to conduct a security assessment of cw3-fixed-multisig smart contracts due their role in internal dao use.

The security engineers involved with the audit are security and blockchain smart contract security experts with advanced knowledge of smart contract exploits.

The purpose of this audit is to achieve the following:

- Ensure that smart contract functions work as intended
- Identify potential security issues with the smart contracts

In summary, Security Dao identified no impactful improvements to reduce the likelihood and scope of risks. Minor improvements to improve test coverage and upgrade from unmaintained dependencies were found.

- The primary ones are as follows:
- Low test coverage
- Use of unmaintained dependency (from cw-storage-plus sub-dependency)

External threats such as intercontract functions and calls should be validated for expected logic and state and are not covered within the scope of this audit. Only direct rpc contract interaction is considered here not any UI components or frontend wasm interactions are excluded.

# 3. Test Approach & Methodology

Security DAO performed a combination of manual review of the code and automated security testing.

The following phases were used throughout the audit:

1 Research into the architecture, purpose, and use of the platform

2 Manual code review and walkthrough

3 Manual Assessment

of the use and safety for critical rust variables and functions in scope to identify any contracts logic related vulnerability

4 Fuzz Testing

Securitydao fuzzing tool

5 Check Test Coverage

**29.59%** coverage, **1072/3623** lines covered (cargo tarpaulin)

6 Scanning of Rust files for vulnerabilities

```
warning: 1 allowed warning found!
Crate:              serde_cbor
Version:            0.11.2
Warning:            unmaintained
Title:              serde_cbor is unmaintained
Date:               2021-08-15
ID:                     RUSTSEC-2021-0127
URL:                https://rustsec.org/advisories/RUSTSEC-2021-0127
Dependency tree:
serde_cbor 0.11.2
                    criterion 0.3.5
                     cw-storage-plus 0.13.0

warning: 1 allowed warning found
```

# 4. Risk Methodology

## 4.1. Risk Likelihood Scale

| | | | | |
|---|---|---|---|---|
| 1 | 2 Unlikely | 3 Possible | 4 Likely to happen | 5 High |

A low likelihood risk indicates that the likelihood of attack is low because of obscurity or requiring additional exploits to utilize, a possible attack is one that is possible but not an attack method commonly seen in the wild or well-known, and high risk likelihood represents an exploit extremely likely to be used, readily apparent, or commonly been used in the past against similar systems

## 4.2. Risk Impact Scale

| | | | | |
|---|---|---|---|---|
| 1 Low | 2 Limited | 3 Impactful | 4 Critical | 5 Severe |

In the context of smart contracts, a low risk impact might be something associated with limited scope or a preventive best practice, an impactful risk may result in large loss of funds but not in a systematic way, and a severe risk impact could result in substantial loss of funds in a systematic way.

# 5. Scope

## 5.1. Cosmwasm smart contracts

The primary target for the audit is cw3-fixed-multisig.  User interface and cross contract messages are considered out of scope for this work.

### Repo

### Public

https://github.com/CosmWasm/cw-plus/tree/main/contracts/cw3-fixed-multisig/src

### Commit hash:

bb4bac178cf5e499b4f65f78ef8af0b4a335ccd8

# 6. Action plan

No major vulnerabilities were identified. Recommendations focus on improvements to the codebase. Validate and lowercase addresses

## Low Effort Low Impact

(SEC - 11) Increase Test Coverage

## High Effort Low Impact

(SEC - 12) Find new maintainer for serde_cbor

# 7. Assessment Summary and Findings Overview

## Findings and Tech Details

# (SEC-11) Increase Test Coverage to > 50%

## Severity Low, Low

## Description

Test coverage is fairly low for such an important contract (<50%)

Code Location

https://github.com/CosmWasm/cw-plus/tree/main/contracts/cw3-fixed-multisig/src

## Risk level

The risk likelihood is **1: low** and the impact is **1: low**

## Recommendation

Add further test coverage for contract.

## Remediation plan

- No further immediate remediation is required.

# (SEC-12) Find Maintainer for serde_cbor

**Severity Low, Low**

## Description

Lack of maintainer for serde_cbor

**Code Location**

https://github.com/CosmWasm/cw-plus/blob/main/contracts/cw3-fixed-multisig/src/contract.rs

## Risk level

The risk likelihood is **1: low** and the impact is **1: low**

## Recommendation

Find a new maintainer for this crate or consider creating a new forked crate to use going forward.

## Remediation plan

- Upgrade contract to use latest version of cosmwasm-std library

Security DAO