AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

TABLE OF CONTENTS

contact@auditmos.com

AUDITMOS

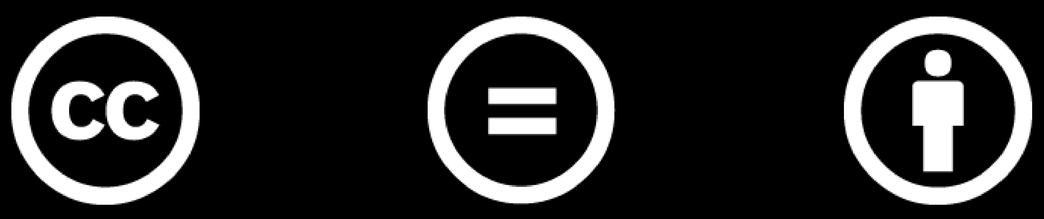
Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

LICENSE



contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

DISCLAIMER

contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

APPROACH AND METHODOLOGY

PURPOSE

1. Determine the correct operation of the protocol, according to the design specification.
2. Identify possible vulnerabilities that could be exploited by an attacker.
3. Detect errors in the smart contract that could lead to unexpected behavior.
4. Analyze whether best practices were followed during development.
5. Make recommendations to improve security and code readability.

CODEBASE

Repository	https://github.com/Team-Kujira/orca-vault
Branch	main
Commit hash	11b9f8d29a8a378838f2dfc45dcd2d25c35b34c5

METHODOLOGY

1. Reading the available documentation and understanding the code.
2. Doing automated code analysis and reviewing dependencies.
3. Checking manually source code line by line for security vulnerabilities.
4. Following guildlines and recommendations.
5. Preparing this report.

contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

DESCRIPTION

Issues Categories:

Severity	Description
CRITICAL	vulnerability that can lead to loss of funds, failure to recover blocked funds, or catastrophic denial of service.
HIGH	vulnerability that can lead to incorrect contract state or unpredictable operation of the contract.
MEDIUM	failure to adhere to best practices, incorrect usage of primitives, without major impact on security.
LOW	recommendations or potential optimizations which can lead to better user experience or readability.

Each issue can be in the following state:

State	Description
PENDING	still waiting for resolving
ACKNOWLEDGED	know but not planned to resolve for some reasons
RESOLVED	fixed and deployed

contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

AUDIT SCOPE

1. getting to know the project	✓
2. research into architecture	✓
3. manual code read	✓
4. check of permissions	✓
5. identify common Rust vulnerabilities	✓
6. test coverage	✓
7. static analysis	✓

contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

FINDINGS

Finding	Severity	Status
#1 - double check of permissions	LOW	RESOLVED

contact@auditmos.com

AUDITMOS

Secure your space

KUJIRA (ORCA-VAULT)

audit / code review report

January 24, 2022

#1 - DOUBLE CHECK OF PERMISSIONS

Severity	Status
LOW	RESOLVED

RECOMMENDATION

Double check of permissions is not necessary as far as function update_config is used only in one place.

At the time of writing function check_owner is checking caller permission as well as:

```
if deps.api.addr_canonicalize(info.sender.as_str())? != config.owner {
return Err(ContractError::Unauthorized {});
}
```

in update_config.

PROOF OF SOURCE

https://github.com/Team-Kujira/orca-vault/blob/11b9f8d29a8a378838f2dfc45dcd2d25c35b34c5/src/contract.rs#L103

https://github.com/Team-Kujira/orca-vault/blob/11b9f8d29a8a378838f2dfc45dcd2d25c35b34c5/src/contract.rs#L534

contact@auditmos.com

auditmos.com

Secure your space

contact@auditmos.com