G0 Group

Security Review of

Orca Protocol

October 2021

Orca Protocol / October 2021

Files in scope

Following files in:

https://github.com/orcaprotocol/contracts/tree/0f77d81d1b8818e7a3ac633c9d71e5267646b318/contracts/

- Controller.sol
- ControllerRegistry.sol
- MemberToken.sol
- SafeTeller.sol

Current status

All issues have been fixed by the developer. There are no known issues in the relevant contracts in https://github.com/orcaprotocol/contracts/tree/a14e76a9ba36456f5c1bcbbaf35e1443e4fb1d7e/contracts/

Issues

1. Controller.updatePodAdmin sets safeAddress instead of podAdmin

type: incorrect implementation / severity: major

Function `Controller.updatePodAdmin` is supposed to allow safe owners to set the`podAdmin` address for their safe, instead it sets the `safeAddress` mapping that controls which tokens control which safes.

status - fixed

Issue has been fixed and is no longer present in

https://github.com/orcaprotocol/contracts/tree/a14e76a9ba36456f5c1bcbbaf35e1443e4fb1d7e/contracts/

2. The fact one safe can be associated with multiple podIds leads to unnecessary trust assumptions and potentially corrupted state

type: security / severity: medium

The fact there's no mechanism that prevents different podIds to be associated with the same safe address through the `safeAddress` mapping means that `createPodWithSafe` can be called repeatedly on the same safe, leading to production of multiple incompatible sets of owner tokens for the same safe. Another implication is that owner of `controllerRegistry` can use `Controller.updatePodState` to make themselves `podAdmin` of any registered safe and hijack it.

status - fixed

Issue has been fixed and is no longer present in

https://github.com/orcaprotocol/contracts/tree/a14e76a9ba36456f5c1bcbbaf35e1443e4fb1d7e/contracts/

Notes

- Since `Controller.memberToken` and `Controller.controllerRegistry` can't be changed, they can be immutable.
- `SafeTeller.createSafe` `podId` argument is unused.
- `MemberToken.CREATE_EVENT` is unused.
- `controller.functionCall` is used in `MemberToken._beforeTokenTransfer` instead of using the contract's interface for no obvious reason.
- `context` state variable is not necessary and could be replaced by `address(this)`.