Protocol Audit Report

Version 1.0

Vesko.io

October 19, 2024

2024-10-KLEIDI

Vesko.io

October 19, 2024

Prepared by: Vesko Lead Auditors: - Veselin Vachkov

Table of Contents

Disclaimer

The Veselin's team makes all effort to find as many vulnerabilities in the code in the given time period, but holds no responsibilities for the findings provided in this document. A security audit by the team is not an endorsement of the underlying business or product. The audit was time-boxed and the review of the code was solely on the security aspects of the Solidity implementation of the contracts.

Risk Classification

		Impact		
		High	Medium	Low
Likelihood	High	H	H/M	M
	Medium	H/M	M	M/L
	Low	M	M/L	L

We use the CodeHawks severity matrix to determine severity. See the documentation for more details.

Issues found

Severtity	Number of issues found
High	0
Medium	1
Low	0
Info	2
Gas	1
Total	4

Findings

Medium

M-1: `abi.encodePacked()` should not be used with dynamic types when passing the result to a hash function such as `keccak256()`

Use `abi.encode()` instead which will pad items to 32 bytes, which will [prevent hash collisions] (e.g. `abi.encodePacked(0x123,0x456) => 0x123456 => abi.encodePacked(0x1,0x23456)`, but `abi.encode(0x123,0x456) => 0x0...1230...456`). Unless there is a compelling reason, `abi.encode` should be preferred.

2 Found Instances

- Found in src/utils/Create2Helper.sol Line: 25

```
                    abi.encodePacked(creationCode, constructorParams)
```

- Found in src/utils/Create2Helper.sol Line: 47

```
                    abi.encodePacked(
```

Informational

[I-1] Owner checks twice: once in `RecoverySpellFactory:createRecoverySpell` and once in `RecoverySpellFactory:calculateAddress`

Recommended Mitigation: - Consider refactoring the duplicate check into a private function to maintain DRY (Don't Repeat Yourself) principles.

[I-2]: Unused Imports

Redundant import statement. Consider removing it.

6 Found Instances

- Found in src/Guard.sol Line: 5

```
import {Safe} from "@safe/Safe.sol";
```

- Found in src/InstanceDeployer.sol Line: 12

```
import {Guard} from "src/Guard.sol";
```

- Found in src/InstanceDeployer.sol Line: 15

```
import {calculateCreate2Address, Create2Params} from "src/utils/Create2Helper.sol";
```

- Found in src/Timelock.sol Line: 19

```
import {Safe} from "@safe/Safe.sol";
```

- Found in src/TimelockFactory.sol Line: 4

```
import {calculateCreate2Address} from "src/utils/Create2Helper.sol";
```

- Found in src/deploy/SystemDeploy.s.sol Line: 8

```
import {Timelock} from "src/Timelock.sol";
```

Gas

[G-1] State variable could be declared constant in `SystemDeploy`

Description:

```
bytes32 public salt =
        0
           x0000000000000000000000000000000000000000000000000000000000003afe
           ;
```

Recommended Mitigation: - State variables that are not updated following deployment should be declared constant to save gas.