Smart Contract Audit Report

Introduction

A smart contract security review of the **Airdrop contract** was done by **Piyush shukla**, with a focus on the security aspects of the application's smart contracts implementation.

Disclaimer

A smart contract security review can never verify the complete absence of vulnerabilities. This is a time, resource and expertise bound effort where I try to find as many vulnerabilities as possible. I can not guarantee 100% security after the review or even if the review will find any problems with your smart contracts. Subsequent security reviews, bug bounty programs and on-chain monitoring are strongly recommended.

About **Auditor**

Piyush shukla,is an independent smart contract security researcher. Having found numerous security vulnerabilities in various protocols, he does his best to contribute to the blockchain ecosystem and its protocols by putting time and effort into security research & reviews. Curretly he's Secured to 3 Hacker Rank globally in Smart Contract Security Platform) or reach out on Twitter Piyushshukla__

About **ProtocolName**

explanation what the protocol does, some architectural comments, technical documentation

Observations

Severity classification

Severity	Impact: High	Impact: Medium	Impact: Low
Likelihood: High	Critical	High	Medium
Likelihood: Medium	High	Medium	Low
Likelihood: Low	Medium	Low	Low

Impact - the technical, economic and reputation damage of a successful attack

Likelihood - the chance that a particular vulnerability gets discovered and exploited

Severity - the overall criticality of the risk

Security Assessment Summary

***review commit hash* - Airdrop contract**

***fixes review commit hash* -**

Scope

The following smart contracts were in scope of the audit:

- `Airdrop contract`

Findings Summary

ID	Title	Severity	Status
[H-01]	incorrect Modifier in emergencyWithdraw Function	High	Fixed
[M-01]	LACK OF DISABLEINITIALIZERS	Medium	Acknowledge
[G-01]	use ++i instead i++	low	Fixed
[G-02]	There is no need to initialize `i` to its default value	low	Fixed

Detailed Findings

[H-01] incorrect Modifier in emergencyWithdraw Function

Airdrop contract, there is a discrepancy in the modifier used for the emergencyWithdraw function. The modifier used is whenNotPaused, which allows the function to be executed when the contract is not paused. However, the function is intended to be used for emergency withdrawals when contract is paused , or you should be allowed even when the contract is paused and not paused

Mitigation

```
Replace modifier Whennotpaused modifier with  Whenpaused
 or
Remove both modifier
```

Severity

High

Status

Fixed

[M-01]LACK OF DISABLEINITIALIZERS

Attcaker can re-initailize the function you must be add _disableInitializers()

Mitigation

constructor() { _disableInitializers(); }

Severity

Medium

Status

Acknowledge

[G-01] use ++i instead i++ in function airdropToken loop

use ++i instead i++ //

Severity

low

Status

Fixed

[G-02] There is no need to initialize `i` to its default value in function airdropToken loop

uint i; instead uint i=0;

Severity

low

Status

Fixed

Powered by Safe Edges