V1.0

HodlCoin

Audit

Dean Eigenmann - Zero Knowledge Labs

December 25, 2017

Disclaimer

The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bugfree status. The audit documentation is for discussion purposes only.

Overview

The audit was performed on all smart contracts found the HodlCoin contracts repository (https://github.com/Arachnid/hodlcoin), the contracts are found in the directory: contracts. The commit hash this audit was performed upon is: 63279761b8980e07d588b9afebb6ffa514814044

Audit Results

The smart contracts developed by the Nick Johnson have been kept to a very high standard, due to the logical flow of how the program works, no security related bugs could be uncovered.

Hodlcoin.sol

This contract contains the main logic for the program, this includes miniting as well as burning tokens.

The contract inherits from previously audited StandardToken contract written by the Open Zeppelin team.

Default function

The default function calls the deposit function.

Value

This function returns the passed amount, multiplied by the balance of the contract divided by the value of the totalSupply.

Deposit

This function mints tokens proportionally to the sender. This will increase the totalSupply and the balance of the sender.

If the totalSupply is equal to zero, then the amount of tokens minted is equal to the amount of ether sent, multiplied by the MULTIPLIER constant.

Otherwise, the amount is calculated by multiplying the totalSupply by the sent amount, and then dividing it by the contract balance subtracted by the sent amount. Additionally a 2 percent fee is subtracted.

The subtracted fee is proportionally distributed by the calling of the withdraw function, this makes it beneficial to deposit ether early and withdraw it later than others.

Suggestions

1. Lines 31-37 could be moved into a seperate function, this would remove the need for an else clause by using early returns instead.

Withdraw

This function reduces the totalSupply and balance of the sender by the amount of tokens specified, it then sends an amount of ether back to the sender. This amount is calculated by the value function.

Suggestions

1. Use an explicit type instead of var on line 49.