Assure DeFi®

THE VERIFICATION **GOLD STANDARD**

Security Assessment

VirtueFinance

Date: 12/06/2025

Audit Status: FAIL

Audit Edition: Advanced

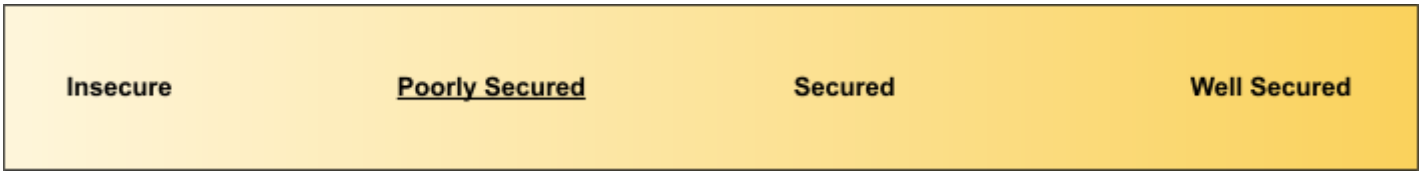
Risk Analysis

Vulnerability summary

Classification	Description
High	High-level vulnerabilities can result in the loss of assets or manipulation of data.
Medium	Medium-level vulnerabilities can be challenging to exploit, but they still have a considerable impact on smart contract execution, such as allowing public access to critical functions.
Low	Low-level vulnerabilities are primarily associated with outdated or unused code snippets that generally do not significantly impact execution, sometimes they can be ignored.
Informational	Informational vulnerabilities, code style violations, and informational statements do not affect smart contract execution and can typically be disregarded.

Executive Summary

According to the Assure assessment, the Customer's smart contract is **Poorly Secured.**



Scope

Target Code And Revision

For this audit, we performed research, investigation, and review of the VirtueFinance contracts followed by issue reporting, along with mitigation and remediation instructions outlined in this report.

Target Code And Revision

Project	Assure
Language	Solidity
Codebase	https://etherscan.io/address/0xbcc711e8ed39b3ca38364f4135571c9877155875#code
Audit Methodology	Static, Manual

Attacks made to the contract

In order to check for the security of the contract, we tested several attacks in order to make sure that the contract is secure and follows best practices.

Category	Item
Code review & Functional Review	• Compiler warnings. • Race conditions and Reentrancy. Cross-function race conditions. • Possible delays in data delivery. • Oracle calls. • Front running. • Timestamp dependence. • Integer Overflow and Underflow. • DoS with Revert. • DoS with block gas limit. • Methods execution permissions. • Economy model. • Private user data leaks. • Malicious Event log. • Scoping and Declarations. • Uninitialized storage pointers. • Arithmetic accuracy. • Design Logic. • Cross-function race conditions. • Safe Zeppelin module. • Fallback function security. • Overpowered functions / Owner privileges

AUDIT OVERVIEW

HIGH

1. Re-Entrancy via addLiquidityETH

Function: _transfer

Issue: No re-entrancy guard on owner's addLiquidity, which calls external router payable function.

Recommendation: Add lockSwap-style mutex to addLiquidity, or use OpenZeppelin's ReentrancyGuard.

2. Centralized Control & Single Point of Failure

Function: All admin

Issue: Owner and taxAddress hold unilateral power (e.g., modifying exemptions, draining funds).

Recommendation: Consider multisig or timelock for sensitive operations; allow taxAddress rotation via governance.

3. Re-opening Trading Abuse

Function: openTrading

Issue: Owner can repeatedly call openTrading, resetting blockLaunch and manipulating anti-bot tax.

Recommendation: Restrict openTrading to one-time use or emit event and block further calls.

MEDIUM

1. Dynamic High Tax (80%)

Function: _transfer

Issue: Anti-bot logic allows tax >20%, bypassing setTax cap, potentially locking trades.

Recommendation: Enforce an absolute maximum (e.g. ≤50%) on dynamically computed tax or remove 80% clause.

2. Front-Running & Sandwich Attacks

Function: _transfer

Issue: Concentrated swap logic on sells (swapTokensEth) can be MEV-exploited.

Recommendation: Randomize swap thresholds or split swaps over blocks; consider time-weighted average pricing.

3. Race Condition on Limit Updates

Function: Limit sestters

Issue: changing maxWallet/maxTransaction in-flight can block user transactions unexpectedly.

Recommendation: Emit events on limit changes; consider grace period before new limits take effect.

1. Parameter Overflow & Rever

Function: triggerSellCA

Issue: Multiplying amount * 10**decimals can overflow and revert.

Recommendation: Validate amount bounds before scaling; use SafeCast.

2. Parameter Overflow & Rever

Function: Unchecked Return Value

Issue: ERC-20 transfer return value ignored, risking silent failure.

Recommendation: Use OpenZeppelin's SafeERC20 to handle transfer return values and revert on failure.

1. Miner manipulation

Issue: Reliance on block.number for anti-bot can be skewed by miners.

Recommendation: Consider using block.timestamp or time-based window instead of single block number.

Technical Findings Summary

Findings

Vulnerability Level	Total	Pending	Not Apply	Acknowledged	Partially Fixed	Fixed
High	3					
Medium	3					
Low	2					
Informational	1					

Assessment Results

Score Results

Review	Score
Global Score	**60/100**
Assure KYC	Not completed
Audit Score	60/100

The Following Score System Has been Added to this page to help understand the value of the audit, the maximum score is 100, however to attain that value the project must pass and provide all the data needed for the assessment. Our Passing Score has been changed to 84 Points for a higher standard, if a project does not attain 85% is an automatic failure. Read our notes and final assessment below. The Global Score is a combination of the evaluations obtained between having or not having KYC and the type of contract audited together with its manual audit.

Audit FAIL

Following our comprehensive security audit of the token contract for the VirtueFinance project, the project did not fulfill the necessary criteria required to pass the security audit.

Disclaimer

Assure Defi has conducted an independent security assessment to verify the integrity of and highlight any vulnerabilities or errors, intentional or unintentional, that may be present in the reviewed code for the scope of this assessment. This report does not constitute agreement, acceptance, or advocating for the Project, and users relying on this report should not consider this as having any merit for financial adVirtueFinance in any shape, form, or nature. The contracts audited do not account for any economic developments that the Project in question may pursue, and the veracity of the findings thus presented in this report relate solely to the proficiency, competence, aptitude, and discretion of our independent auditors, who make no guarantees nor assurance that the contracts are entirely free of exploits, bugs, vulnerabilities or deprecation of technologies.

All information provided in this report does not constitute financial or investment adVirtueFinance, nor should it be used to signal that any person reading this report should invest their funds without sufficient individual due diligence, regardless of the findings presented. Information is provided ‘as is, and Assure Defi is under no covenant to audit completeness, accuracy, or solidity of the contracts. In no event will Assure Defi or its partners, employees, agents, or parties related to the provision of this audit report be liable to any parties for, or lack thereof, decisions or actions with regards to the information provided in this audit report.

The assessment serVirtueFinances provided by Assure Defi are subject to dependencies and are under continuing development. You agree that your access or use, including but not limited to any serVirtueFinances, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies with high levels of technical risk and uncertainty. The assessment reports could include false positives, negatives, and unpredictable results. The serVirtueFinances may access, and depend upon, multiple layers of third parties.