Audit Report - 0c770343-59a5-412b-b127-6c57b829acab

Vulnerabilities

MEDIUM - Line 34

Issue: Use of `block.timestamp` can be manipulated by miners

Recommendation: Consider using a more secure source of time or a mechanism that doesn't rely on miner-controlled variables.

LOW - Line 27

Issue: Potential for unbounded loop due to `userAuditHashes[msg.sender].push(codeHash);`

Recommendation: Consider implementing a mechanism to limit the number of audits per user or use a more gas-efficient data structure.

LOW - Line 47

Issue: Potential for returning an empty array when a user hasn't performed any audits

Recommendation: Consider handling this case explicitly or documenting the expected behavior.

Gas Optimizations

Estimated Gas Usage: 2,50,000 gas

Line 27: Using a more gas-efficient data structure like a linked list or a fixed-size array for `userAuditHashes` could reduce gas costs for storing new audits.

Estimated Savings: 5,000 gas

Line 34: Caching `block.timestamp` in a local variable can save gas on repeated accesses.

Estimated Savings: 100 gas

Line 40: Using `AuditEntry memory entry = audits[user][codeHash];` directly in the return statement could save gas by avoiding an extra memory allocation.

Estimated Savings: 200 gas