

“StorageVictim” Smart Contract: Fixed Assessment

1. Scope

After the initial assessment for security flaws, the recommended changes were made to the smart contract called “StorageVictim”. Click on the following link to view the fixed code:

<https://github.com/jsanders108/StorageVictim-Audit/blob/main/Fixed/FixedStorageVictim.sol>

The following process was followed to complete this security audit on the fixed contract:

- a. The static analysis tool called “Slither” was run on the contract to identify any red, yellow, or green issues that still exist.
- b. The contract was tested manually to determine if any intended functionality was not working correctly after code changes were implemented.

2. Findings

2.1 Static Analysis Testing: Slither

Red Issues

NONE

Yellow Issues

NONE

Green Issues

Issue #1)

Per the initial assessment’s recommendation, the contract was compiled using Solidity version 0.8.16. According to Slither, however, version 0.8.16 is not recommended for deployment—even though it is recommended in the reference document below:

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity>

Recommendation: Because this is a green-level issue, it is likely okay for the contract to be deployed using Solidity version 0.8.16. It is possible that the reference documents have not been updated.

Issue #2)

Per the initial assessment's recommendation, the owner of the contract was set to msg.sender upon deployment by using a constructor:

```
address owner;  
  
constructor() {  
    owner = msg.sender;  
}
```

According to Slither, the state variable "owner" should be declared as "immutable".

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable>

Recommendation: Set the "owner" variable as "immutable":

```
address immutable owner;  
  
constructor() {  
    owner = msg.sender;  
}
```

2.2 Final Manual Review & Testing

No issues found.