

Security Issue 1: There was no backup policy for my code project and a VCS was not being utilized.

Solution:

I uploaded my project and project files into a GitHub repo. The CLion IDE I use to make maintain and continue the project was also used to link directly to the repository to make any further changes to GitHub. Since I have pushed my codebase to a GitHub repository, I am always able to clone the repo from GitHub to restore my code in the case my computer is lost/stolen, or my hard drive no longer functions so I can retrieve my data. GitHub also allowed for all the libraries and CMake files that were included in the project to be uploaded and cloned when needed.

<https://github.com/josefgcu/snakegame>

Security Issue 2: All functions were in the main file of the program, giving access to all the functions and how the game works just by accessing the snakegame.cpp main source file.

Solution:

I broke up the code into classes for the snake, food, and wall allows for better organization and encapsulation of code, making it easier to maintain and update. Also, separating the code into distinct classes can help limit the amount of security vulnerabilities, as any issues that arise will be confined to a specific class rather than affecting the entire program. By limiting access to the functionality of each class through their unique modifiers, it can help ensure that any data or functions within a class can only be accessed in ways that are safe and secure.

<https://github.com/josefgcu/snakegame/tree/master>

Future Issues:

1. Add more error handling or exceptions to see if anything were to fail within the functions of each class to catch if any are tampered with.

Priority: High

Ease of fix: Moderate

2. Add restrictive access control to certain files that way they can be hidden from any type of possible tampering users (change repository visibility)

Priority: Medium

Ease of fix: Moderate/Difficult

3. Provide more security of actual computer code is stored on (add protected password and encrypted files)

Priority: Mid

Ease of fix: Moderate

4. Check third party libraries used to make the code is not compromised and is completely up to date.

Priority: High

Ease of fix: Moderate