Kevin Zhang

Application Security CS GY 9163

**Homework 1: Beware if Geeks Bearing Gift Cards**

On NYU Classes, submit a link to your GitHub repository. The repository should be **private**, and you should add the instructor/TA's GitHub account as a contributor to give them access for grading.

For this section, your instructor is: Kevin Gallagher, GitHub ID `**kcg295**`

Your TA is: Evan Richter, GitHub ID `**evanrichter**`

The repository should contain:

**Part 1: Setting up Your Environment**

Relatively self-explanatory. The steps for Part 1 are detailed below:

GitHub Set-up

* Git-VCS has been installed.
* Git Bash has been installed (Windows)
* Git signed commits have been set-up
* GitHub Account created – kzhang112
* GitHub-Repo created (“AppSec\_1.1”) – set to “private”

Travis CI

* Travis-CI account created and tied to GitHub
* .travis.yml file created
* Iterations (changes of code) Travis builds until build is successful

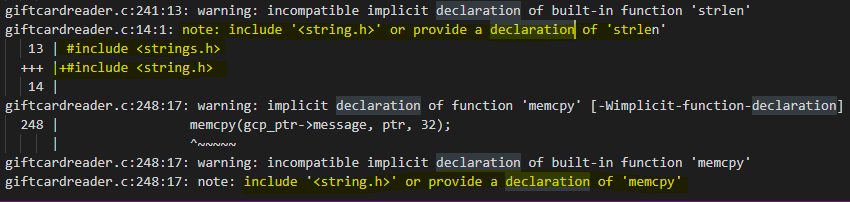
**Part 2: Auditing and Test Cases**

Set-up

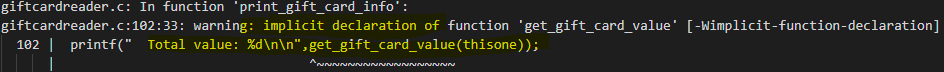
* Files added to GitHub Repo
  + “*examplefile.gft*”
  + “*giftcardexamplewriter.c*”
  + “giftcard.h”
  + “*giftcardreader.c*”
  + “*Makefile”*

After adding the above files to my GitHub Repo (“kzhang112/AppSec1.1”), I attempted to run both “*giftcardreader.c”* and “*giftcardwriter.c*” using my Visual Studio Code compiler (I ran the program through the Mingw-x32 for Windows). “*Giftcardwriter.c*” did not return any errors prior to running the program, and I was able to print out an example gift card (similar to the gift card provided in the original repository).

However, “*giftcardreader.c”* ran into an initial error indicating that I needed to include a string header.



After adding in “#include <string.h>”, I encountered another error.



By adding in an extra int for get\_gift\_card\_value prior to the code block where we see the error, it gives the compiler information that can be used later in the program (e.g. in this case, on line 102).

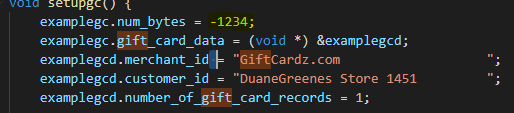


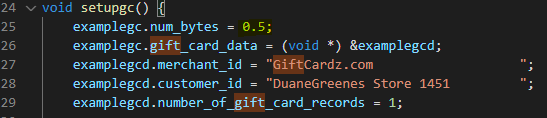
Finally, I was able to build and run the file on the example giftcard that I had just created.

Find Flaws

* “*Crash\_1.gft*” created







If the num\_bytes portion of the giftcard is populated with a negative number, or a decimal, then the program crashes.

* “*Crash\_2.gft*” created
* “*Hang.gft*” created
* *“Bugs.txt”* created, describing the above issues
* Bugs highlighted in “*Bugs.txt*”fixed in “*girfcardreader.c*”

**Part 3: Fuzzing and Coverage**

Total points: 100

Part 1 is worth 20 points:

* 10 points for signed commits
* 10 points for Travis configuration

Part 2 is worth 40 points:

* 10 points for your test cases and fixes
* 20 points for the bug writeup
* 10 points for Travis regression testing

Part 3 is worth 40 points:

* 10 points for handwritten tests
* 10 points for fuzzer-generated tests
* 10 points for your code fixes
* 10 points for writeup

## What to Submit

On NYU Classes, submit a link to your GitHub repository. The repository should be **private**, and you should add the instructor/TA's GitHub account as a contributor to give them access for grading.

For this section, your instructor is: Kevin Gallagher, GitHub ID kcg295.

For this section, your TA is Evan Richter, GitHub ID evanrichter.

The repository should contain:

* Part 1
  + Your .travis.yml
  + At least one signed commit
* Part 2
  + A directory named part2 that contains crash1.gft, crash2.gft, hang.gft, and bugs.txt
  + An updated .travis.yml that runs your tests
  + A commit with the fixed version of the code (if you like, this commit can also contain the files mentioned above)
* Part 3
  + A directory named part3 that contains cov1.gft, cov2.gft, fuzzer1.gft, fuzzer2.gft, and writeup.txt
  + An updated .travis.yml that runs the new tests
  + A commit with the fixed version of the code (if you like, this commit can also contain the files mentioned above)