CSC 4980 / 6980 Blockchain & Applications

Assignment 2

Due Date: 11:59 pm, February 25th, 2019

In this assignment the student is expect to code a Python 2.7 program that breaks SHA1 hashes in a brute force manner. This program will have the following requirements:

1) The user will input, as an argument, the hash value to the program.
2) You should use a password list. I want you to use: https://raw.githubusercontent.com/danielmiessler/SecLists/master/Passwords/Common-Credentials/10-million-password-list-top-1000000.txt
   However, you are free to use any additional ones (this file is the minimum required). You should copy them locally for ease of use.
3) Your program should count the number of tries before reaching a solution, if found.
4) This program should be done using standard python libraries (hint: hashlib). Ideally no additional libraries should be used. If you are using extra libraries, please provide clear documentation on how to install them.
5) Your program should include a README.md file with detailed instructions on how to run the program and the solution to all the exercises in this assignment. This file should be properly formatted using GitHub Markdown language.

Your program will be graded for correctness, completeness and its ability to crack hashes. If the program does not work, you will receive a 0 on the assignment. The program itself is worth 100 points. Answers to the following questions will NOT count if your program cannot produce the solutions you are claiming, so be sure it works.

For the provided SHA1 hashes, please time how long it takes to break the hash (find a match) and the actual clear text password behind the hash value. In other words, what is the actual password before hashing for the following hashes:

a) **(20 points) testing program hash:** b7a875fc1ea228b9061041b7cec4bd3c52ab3ce3
b) **(25 points) medium hacker hash**: 801cdea58224c921c21fd2b183ff28ffa910ce31
c) **(30 points) leet hacker hash**: ece4bb07f2580ed8b39aa52b7f7f918e43033ea1
   Hint: The salt term here is: f0744d60dd500c92c0d37c16174cc58d3c4bdd8e this is concatenated before hashing with another word to produce the salted hash.
d) **(40 points) graduate student**: 34302959e138917ce9339c0b30ec50e650ce6b40
   Hint: This hash constitutes two terms separated by one space

BONUS (60 points): Write an additional program that improves the performance of the basic brute force program you have just created. This 'improved' program requires very minor tweaks to work a lot faster.

What to turn in:
Each student is responsible to turn in their own code. Plagiarism will be heavily prosecuted, so please do not risk it. No two people code alike and this should be reflected in the programs.

1) You should upload all running code and the README.md file to your own GitHub repository. Yes, only GitHub, no other services will be accepted.

2) The README.md file should contain instructions on how to run your code and the solution to the provided exercises. If you wrote the improved program, also add instructions on how to run that program and the solution to the exercises using that program as well.
3) The README.md file should be properly created using GitHub markdown (https://guides.github.com/features/mastering-markdown/) it should have the student's name clearly displayed somewhere.

Upload to iCollege the downloaded ZIP file of your GitHub repository. There is a clone/download button on the main page. You should also submit the URL of your GitHub repository in the submission comments.

IMPORTANT: Failure to use GitHub and provide the indicated README.md file with instructions and answers will lead to a zero grade in the assignment.