CMPSCI 177 - Computer Security
Fall 2013
Fifth Homework - Password Security

Due: 20 NOV 13 at 2:00pm

I have brought up four bogus student accounts in the CSIL lab. They have the following login names: fooshbal, gorbels, cdidit, and lapisnek.

The encrypted passwords for these accounts are provided on the CS177 solutions page. Your task is to attempt to compromise these accounts by guessing their passwords. You may use any technique you want; including dictionary search, exhaustive search, intelligent guessing, etc. Some of the passwords will be easier to guess than others.

You are to compromise as many accounts as possible.

After you have compromised an account email a message to me from the account giving me YOUR name. My login is kemm.

WARNING:
DO NOT include the password in your message to me.
DO NOT change anything in the account.
DO NOT send me spoofed email (you will be docked points if you do).

These are the ONLY accounts you are to attempt to compromise.

You should be courteous to other CSIL users. DO NOT run your programs on multiple machines in the lab. If you do, you risk having your account closed.

As you attempt to complete this homework, you will find that there are cracker tools on the web that are very efficient at cracking weak passwords. You are welcome to use these tools. Unfortunately, if you use these tools as your only method of cracking the passwords, you will not learn much about crackers and how they work. Therefore, I want you to build your own cracking tools, preferably using C or Python. Your cracking program **must** include benchmarking, to determine the speed and effectiveness of your program. You should determine the number of hashes computed per minute for each of the password hashing algorithms used by the 8 users.

Here is a list of things that you may want to implement in your cracker program:
1. Check the words from a given dictionary file (one word per line)
2. Check the reverse of the words in a dictionary file
3. Check different characters of the word in both uppercase/lowercase
4. Check combinations of 2 words in a dictionary
5. Perform a brute force exhaustive search

This list is not meant to be complete, nor is it guaranteed to find the passwords for all four accounts. Use your imagination and the hints given in the text and in class to build your search techniques. C programs should be compiled in CSIL using gcc to assure that the TA will be able to do the same.

What you are to turn in to me is your C or Python source code, a README file that describes

> (a) how your tools work - be sure to include a Makefile and give the commands to execute your tool
> (b) how you arrived at the passwords that you successfully cracked - both those that you cracked using your tools and those that you cracked using net tools but failed with your tools, and
> (c) what techniques you tried to crack the passwords for which you were not successful. We will run your code against the passwords that you claim to have cracked with your code, as well as other passwords of similar form.
> (d) the results of your benchmarking: The number of hashes your program tries in a minute, for each of the two password hashing algorithms used. You should run your program on 1) a CSIL machine and 2) The fastest machine you have legitimate access to and report the specifications and statistics from each machine.

Instructions for turning in your homework are as follows:
1. Create a directory that is the same as your CS login. For example, user John Doe --whose account is jdoe-- would do:

```
% mkdir jdoe
```

2. Put the source code and README under the directory created above. Do not include binary code in this directory.
3. Connect to csil.cs.ucsb.edu
4. Execute the turnin program. For example, user jdoe would execute:

```
% turnin hw5@cs177 jdoe
```

You can execute turnin up to 10 times. Earlier versions will be discarded. The timestamp of turnin has to be before the due date.
Reminder: Don’t forget to send me (kemm@cs) an email from each of the accounts at the time that you compromise them.

Good Luck!