CMSC 491: Introduction to Cyber Security

Assignment 2

October 2, 2015

Due October 16, 2015 at 11:59pm through blackboard submission. In this assignment, you are expected to work individually. You may use any sources that you want, but you must cite them. The submission should be a zip file named "A2_ < *yourname* >.zip", which contains a written instruction on how to run your code for question 1 and question 2 in a PDF file named "A2.pdf", and all source code files plus any other files required to run the code. Source code should be well documented for a high score. Please include the author's name in the code. My office hours are M/W 3:30pm - 5:30pm in E4234. You can also email me if you have questions.

1 CRACK PASSWORDS

(25 pts) Implement a password cracking program to find the passwords of 5 users offline. The login method is provided in Login.class. Please write a Java program A2Q1.java, so that:
(1) The program can crack the passwords for "user1", "user2", ..., "user5".
(2) Run the program with user names as input and output the password of the user, number of tries, and time used in cracking. For example,
> java A2Q1 user1
the password for user1 is: xxxxx
number of tries: 89947
time used: 29 seconds
Hint: All passwords are 8 characters long. Through some social engineering effort, we know that the password for user1 is a birth date; the password for user2 is a person's name; the password for user3 is a few words concatenated; the password for user4 is simple; the password for user5 is popular. Your program should terminate automatically if run for more than 10 minutes without a successful guess. Note that the Login.class for testing may be different than we sample Login.class.

2 SECURE PASSWORDS

(25 pts) Implement a program to help users to choose secure passwords by rejecting simple passwords and providing hint information. Please write a Java program A2Q2.java for this purpose. The criteria of a good password include:
(1) At least 8 characters long and contains at least 1 letter.
(2) Can not be only the combination of dictionary words.
(3) Can not be too simple (with patterns), for example, abcd1234 is as simple as a1, and aaaaaaaa is as simple as a. If a string is as simple as two characters then it is too simple.
(4) Run the program A2Q2 and it will print a message first to ask the user to input a password. After the user input a password, it will either accept the password or reject with a hint message. If the password is rejected, it will ask the user to try again. For example,
> java A2Q2
Please choose a password:
yam
No, the password has to be at least 8 characters long, Please choose a password:
19980706
No, the password has to contain at least 1 letter, Please choose a password:
Ihateyou
No, the password cannot be the combination of dictionary words, Please choose a password:
babababa
No, the password is too simple, Please choose a password:
killme101
Congratulations! Your new password is set.

3 BONUS

(5 pts) For question 1, please crack the last password for user user6. The only hint is that the user6 is an old lady who has trouble to remember a complex password.
> java A2_bonus user6
the password for user6 is: xxxxx
number of tries: 10996
time used: 20 seconds