**Hsiang Lo**

**CS 370 Introduction to Security Week 4: Problem Set 4**

Instructor Name: Rakesh Bobba

# Introduction

The purpose of this assignment is to help you gain a better understanding and insight into user authentication concepts and the pros and cons of password-based authentication covered about in Week 4.

Before beginning make sure you have watched the lecture videos on the following and completed the associated practice quizzes.

* User Authentication
* Passwords: Pros and Cons I
* Passwords: Pros and Cons II
* Passwords: Pros and Cons III

Also make sure you have read this week’s assigned reading from the textbook.

# Questions

Please answer the questions below.

## User Authentication and Passwords

Q1[10 pts]: You are designing a password system with randomly selected passwords. The alphabet for the passwords is the set of alphanumeric characters in English both upper and lower case and the integers 0-9.  You are told that the attacker can make 250,000 guesses each minute.

1. If the passwords are 7 characters long, how long until the attacker has a 50% probability of correctly guessing user’s passwords in an offline attack.

This questions refers to Anderson’s formula which is P = (TG/N) where

* + - P is the probability of guessing a password in a specified period of time
    - G is Number of guessing test in a 1 time unit
    - T is number of time units
    - N is the number of possible password

Given that formula, we can calculate the time as,

P = TG/N

0.5 = 250000T/62^7

T = 13.4 years

1. How long do the passwords need to be to ensure that the 50% success rate is not reached until after 2 years?

In order to ensure 50% success not of not reaching after 2 years, the password needs to be 7.

Since P = TG/N

0.5 = 24\*60\*365\*2 \* 250000 / 62^ n

Where n is length of guess,

N = 7

1. If the users select their own passwords, does this affect the relevance of your calculations from parts (a) and (b)?  Explain your answer.

Yes this is because when a user choose their own password, it would be more likely to be a passphrase or a non-random pattern that would be more likely to be cracked. Anderson’s formula ensure uniform distribution and random selection, which are two key factors that allows it to be as powerful. If the user chooses their own password, it would lead to easier crack since it would be a weaker password.

Q2 [4 pts]: iPhone 6 includes a fingerprint scanner which the user can choose (not) to use. Do you think activating fingerprint scanning would increase the security of the cellphone? Why or why not?

Given that the technology of fingers scanning won’t be possible to crack, then I would say yes, it is much safer as it allows duo authentication. On the other hand, if it’s not, then no, it won’t really help if adversary can get their hand on it.

Q3 [3pts]: Bloom filter is an efficient way to preemptively reject bad passwords with high efficiency, but it has a false positive rate (incorrectly rejecting good passwords). What can you do to decrease the chance of a false positive?

Bloom filter is an amazing idea that increases password security. While it is good at rejecting bad password, it does sometimes allow good password to be rejected to. I believe the best way to do this is to ensure an extra layer of check which can ensure that the previous computation was correct. This can help it both better reject bad passwords and better accept good passwords.

Q4 [3 pts]: Why will a bloom filter never give a false negative (accept a bad password)?

This is due to the fact that Bloom have a dictionary of bad passwords. So when a user enter a bad password which match up with Bloom’s list of bad passwords, it can immediately reject it and prevent it from being used. This is also because it uses a hash function, and so each character checked with be checked against the hash

Q5 [3 pts]: It is common practice not to store user’s password in clear text. However, if an attacker has seized control of the password database, he is likely already capable of modifying any user data on the site as an administrator. Why bother hashing the passwords then?

While it is true that the adversary can crack the system and then be able to modify any user data on the site as an administrator. The hashing of the password provides an extra layer of safety which in the event that the stored version of the password is stolen, it cannot be openly read. I

Q6 [3 pts]: It is common practice to salt the user’s password in addition to hashing. What attack does this practice prevent?

It prevents dictionary attack and yes it is common. This is due to the fact that salt is capable of randomizing the password hash key, which make it more difficult for adversary to use the same dictionary key to hack into user accounts.

Q7 [4pts]: Does a “salt” used in password hashing need to be kept secret? Why or why not? Compare and contrast “salts” and “initialization vectors (IVs)” used in CBC encryption mode.

Salt used in password hashing does not need to be kept secret since its stored with the hashed key. And since it randomize the output of hash, it makes attack on dictionary attack harder and finally salts and initialization vectors are but not reusable, since they are meant to use random password/hash at all time. And they both randomize the output of the hash/cipher.

# Submission Details

Submit a PDF file with the questions and your corresponding answers.

The assignment is worth 30 points. It is due Saturday of Week 4 at Midnight.