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.

a. 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.

   62^7 = 3,521,614,606,208
   62^7 = 250,000T/0.5
   62^7 * 0.5= 250,000T
   1,760,807,303,104 = 250,000T
   7,043,229.2 = T
   7,043,229.2 / 60 / 24 / 365 = 13.4 years

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

   - N = (2*365*24*60) * 250000 / 0.5 = $5.256*10^{11}$
   - $62^6 \leq N$
   - $62^7 \geq N$
   - 7 characters

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

- This does not affect the relevance of the previous calculations, because the formula works the same no matter the length of the 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?

- Using multi-factor authentication always improves the security of a system because the system requires more proof of who the user is in order to be unlocked. If the wrong user tries to unlock it with multi-factor authentication, the system will not unlock.

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?

- Increase the number of hash functions in order to decrease the probability that every single bit in the bloom filter is set to 1.

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

- The bloom filter checks if a password is a member of a set of bad passwords, and it will never say the password is not a member if it is a member of the set.

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?

- Hashing the passwords within a file ensures that the attacker still has to guess the passwords themselves or invert their hashes.

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

- Dictionary attack.

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 does not need to be kept secret. This is because the salt increases the effort that the attacker needs to put in to decrypt a password anyway.
- Salt is different from an IV because, even though both salt and IVs are random, they work in different ways. With an IV used in CBC, the random value is used to allow the same key to be used to encrypt several different things. Salt, on the other hand, is used to allow the same message to be encrypted multiple times and still produce a different hash value each time.

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.