

COM 402 exercises 2019, session 4:

Database security and Password storage

Exercise 4.1

Your application uses an SQL database which stores the names, grades, year of graduation of students.

- What mechanism can you apply to allow user Alice to only read data of students that graduate in 2020 ?

Exercise 4.2

Consider an application that contains medical data. For more security, it uses two different tables, one with the personal data of patients and another one with their medical conditions. Three database users are defined:

1. a user that can only read and write the personal data table
2. a user that can read the personal data and the medical data
3. a user that can read and write the medical data

The first user is used when patients update their personal data. The second user is used when the patients want to see their medical information. Finally, the third user is used when a doctor logs in to update the medical data.

- If any of the three parts of the application was vulnerable to SQL injections, would this allow a patient to read the medical information of another user?
- What is a typical way of preventing an SQL injection?

Exercise 4.3

- Why is transparent data encryption (the DB encrypts before writing to files) better than an encrypted file system (the OS encrypts the content of the files) ?

Exercise 4.4

- Give two reasons why it is important to salt password hashes

Exercise 4.5

You just built a very nice rainbow table that can crack 99% of 8 letter passwords. Compared to a brute-force attack, it uses 1000 times less hash operations to find a password.

You are given a list of ten thousand hashes that need to be cracked.

- Is it going to be faster to crack the passwords with the rainbow table or with a classical brute-force attack?

Exercise 4.6

- How comes that calculating a Windows password hash (based on MD4) is about 200'000 times faster than calculating a Linux password hash based on SHA512?