CS 370: Problem Set 3

Section: TR 10-11:50 am

Total: 150pts Due: 04/19/2016

Instructions:
1. I leave plenty of space on each page for you. If you need more sheet, please attach your work right behind the corresponding problem. Most of the problems are designed for you to think about the models and the principles.

2. The first assignment has two parts, written part (15pts each question) and programming part (45pts).

3. Submission:
(a) If you are doing the homework as a pair, please inform me in advance and designate one person as the contact window
(b) On the due day, please submit a hard copy of the written part at the beginning of the class and submit your code through blackboard.

First Name:

Last Name:

Group ID:

Score: /

Problem 1 Safety Engineering: 7 + 8 pts

In the insulin pump system, the user has to change the needle and insulin supply at regular intervals and may also change the maximum single dose and the maximum daily dose that may be administered.
(a) Suggest three user errors that might occur. You can check the slide with the caption 'Risk classification for the insulin pump'.

(b) Propose safety requirements that would avoid these errors resulting in an accident. You can check the slide with the caption 'Examples of safety requirement'.

Problem 2 Security Engineering: 15 pts

For the Mentcare system, suggest an example of **an asset, an exposure, a vulnerability, an attack, a threat and a control** in addition to those discussed in the slides. For instance, the asset could be a local database stored on nurses laptop or clinic PC.

Problem 3 Security Engineering: 15 pts

Suppose the attacker intends to paralyze your server and deploys the IP address spoofing trick to fool your traffic control software as he notices that you deploy some traffic controller. Let say the attacker fakes 1000 IP addresses and each sends like 1/100 of the total number of requests (call it τ) your server can handle while your controller is set to allow each IP can send at most $(1/10)\tau$ requests . This might paralyze your system. What would you do to protect system if this occurs? Furthermore, can you think of any useful data structure for solving this? And can you come up with another attack to paralyze this strategy you propose?

Problem 4 Resilience Engineering: 15 pts

Explain how the strategies of resistance, recognition, recovery and reinstatement may be used to provide system resilience.

Problem 5 Resilience Engineering: 15 pts

A hospital proposes to introduce a policy that any member of clinical staff (doctors or nurses) who takes or authorizes actions that leads to a patient being injured will be subject to criminal charges. Explain why this is a bad idea, which is unlikely to improve patient safety and why it is likely to adversely effect the resilience of the organization.

Problem 6 Software Reuse: 15 pts

Using the example of the weather station system described in chapter 7 (design and implementation, suggest a product line architecture for a family of applications that are concerned with remote monitoring and data collection. You should present your architecture as a layered model, showing the components that might be included at each level.

Problem 7 SQL: Logic: 5+5+5 pts

You are given the following pseudo code for a trigger (please refer to the DDL.sql for the table structure):
create trigger credits_earned after **update of takes** on (grade)
referencing new row as nrow
referencing old row as orow
for each row
when **nrow.grade != F and nrow.grade is not null and**
(orow.grade = F or orow.grade is null)
begin atomic
update **student**
set tot_cred= tot_cred + (select credits from **course** where course.course_id= nrow.course_id)
where student.id = nrow.id;
end;
(a) What does this trigger do?

(b) If condition **and (orow.grade = F or orow.grade is null)** is removed, what is the potential problem?

(c) If condition **and (orow.grade = F or orow.grade is null)** is changed to **and (orow.grade is null)**, is it the same as the original query in terms of the result? why?

Problem 8 Programming: SQL: 10+20+7+8 pts

For this part, please please use the data schema (DDL.sql) and data (smallRelationsInsertFile.sql) given in Problem Set 2. Please test your queries before turning them in as your queries will be tested to compare the results.

(a) Write the query to create an instructor_audit table that contains the following fields: **transID, uid, presalary, name , changedat** and **action**. $transID$ is the primary key and must start from 1 and increment automatically by 1 whenever a record is inserted. The data types of uid, presalary and name must be consistent with the data types of ID, salary and name in the instructor table while the data type of action can be simply set to varchar(20).

(b) Write a trigger on the instructor table before the update of salary. For instance, if Mozart's salary is updated from 40K to 50K, then this piece of information (Mozart's ID, Mozart's name, old salary, and time when the change is made) must be recorded into the *instructor_audit* table.

(c) Let say we have the following two queries:

Query 1: create view instructor_info as select ID, name, building from instructor, department where instructor.dept_name= department.dept_name;

Query 2: insert into instructor_info values ('69987', 'White', 'Taylor');

(c1) What happens to the underlying tables (**assume** that we do not have the primary key constraint on the department table; **note** that we do have this constraint in the data set though)?
(c2) And why would this be ambiguous?