Secure and Dependable Systems Module: CO-566
Constructor University Date: 2023-03-03
Dr. Jürgen Schönwälder **Due: 2023-03-10**

Problem Sheet #4

Problem 4.1: *compiler hardening options* (1+1+1 = 3 points)

Compilers often provide special options to mitigate some of the problems we discussed in class. For each of the following `clang` options, briefly describe what they do and demonstrate how they help to prevent some of the problems;

a) `-D_FORTIFY_SOURCE=2`

b) `-fsanitize=safe-stack`

c) `-fstrict-vtable-pointers`

Problem 4.2: *SQL injection* (1+1 = 2 points)

A course registration system uses the following database table:

```
CREATE TABLE exams (
        student         INTEGER NOT NULL,
        exam            TEXT NOT NULL)
```

We identify students by a matriculation number and exams by name, a real system would be more complex.

The registrar's office hired a student to implement a web frontend that lists all exams a student is registered for. The backend executes the following query:

```
SELECT exam FROM exams where student={input}
```

The `{input}` part is replaced with the input provided by the user.

a) You forgot to register for the SADS exam in time. Write an SQL injection to register yourself for the SADS exam. Your student number is 4224.

b) The instructor got an exam participation list before you registered yourself via the SQL injection attack and she discovered an inconsistency. You panic and you like to clear all traces by dropping the entire table. What is the SQL injection to achieve this?

Problem 4.3: *CVSS vulnerability scores* (3+2 = 5 points)

Write a Rust program `cvss` that takes CVSS vectors as arguments and returns the CVSS score and severity.

```
$ cvss CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:N
CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:N score 3.8 severity Low
```

a) Write a Rust program that takes CVSS vectors as command line arguments and returns the CVSS score and severity. As a Rust beginner, you are encouraged to use suitable Rust crates to write your program.

b) Your program should have some unit test cases to ensure that your program handles test inputs correctly.