Exercises, set 4

Computer Security

- You do not have to hand this in. You can completely ignore these exercises if you wish.
- These exercises are given for you to check your knowledge during the course.
- These exercises are based on material in Chapters 9-10, 16-17 and 19-20. Some exercises may require you to look for information outside the textbook as well.
- Exam questions can be based on some or none of the exercises.
- The teachers are happy to help you out if you run into problems.

Exercise 4.1 SSL cipher suites are represented by 16-bit numbers. Each number has a corresponding string. Explain the meaning of the SSL cipher suite given by `TLS_RSA_WITH_RC4_128_MD5`.

Exercise 4.2 How is entity authentication implemented in GSM and how does UMTS improve this authentication protocol?

Exercise 4.3 Describe how bad usage of the gets() function in C can be used in a buffer overflow attack.

Exercise 4.4 In this question we will study the inference problem of databases. Assume that a database table named Students contains the information in Table 1. Possible queries are COUNT, SUM and MEAN. The database does not reveal results when a small or large number of people are covered by the query. In our database it has been decided that 4-7 people must be covered by the query. We want to find a sequence of queries that tells us the size of Erika's loan. The obvious query

```
SELECT SUM(Loan) FROM Students WHERE name = 'Erika'
```

will not work because the result, 1, is not in the range 4-7. The query will be ignored. Find a general tracker and a sequence of queries that will give you the information. You are allowed to do 4 queries.

Exercise 4.5 The stream cipher RC4 is used in both SSL/TLS and in WEP. It is "easy" to recover the secret key in RC4 when used in WEP, but it is considered secure when used in SSL. Explain why. What is the difference?

Exercise 4.6 Why do we not encrypt the ESP header when IPsec is used in ESP mode?

Exercise 4.7 Why must the authentication data in the Authentication header be a multiple of 32 bits?

Exercise 4.8 One way to detect buffer overflows is to use a canary.

Table 1: Student information.

Name	Age	Nationality	Loan	Dorm
Adam	22	Sweden	5000	Sparta
Benny	24	Denmark	0	Delphi
Caroline	20	Norway	3000	Ulrikedal
Daniel	20	Denmark	1000	Delphi
Erika	21	Sweden	2000	Sparta
Filippa	25	Sweden	1000	Ulrikedal
Göran	27	Sweden	4000	Ulrikedal
Henrietta	23	Denmark	5000	Sparta
Karin	22	Sweden	0	Ulrikedal
Linda	21	Norway	0	Delphi
Magnus	26	Sweden	2000	Delphi

a) Explain how the canary helps detecting this attack.
b) Strings in C always end with a NULL character, written as '\0'. The buffer overflow vulnerable function strcpy() will copy a string to a destination. The copying is done when the source string ends, i.e., when a NULL character is found. The function will not care if the space allocated for the destination is enough. Explain how we can protect against buffer overflow attacks targeting the strcpy() function, by using a canary that is always NULL.