PART 1

Exercise 1.1

Consider the following protocol:

$$\begin{array}{lllll} 1. & S & \rightarrow & A & : N_S \\ 2. & A & \rightarrow & S & : E_{k_A}(N_S, A) \end{array}$$

The server S sends a nonce N_S to A who encrypts it (together with her identifier A) under key k_A. This key is derived from A's password pwd_A as $k_A = md5(pwd_A)$ and its value is also known by the server. The server decrypts and authenticates A if N_S and A are correct.

1. What security guarantees provides the above protocol?
2. Why is this protocol more secure than just sending to S username A and password pwd_A?
3. Discuss possible attacks based on the fact that the key is derived from a password.

Exercise 1.2

Explain why revealing the big primes p and q in the RSA cipher breaks the private key. Show a numerical example (with small numbers).

Exercise 1.3

In order to securely transmit stream a bytes (e.g. a satellite transmission or a ssh terminal session) it is proposed to independently encrypt each single byte using AES cipher under a fixed key k_s. Discuss why this solution is completely unsatisfactory and discuss possible attacks.

PART 2

Exercise 2.1

Consider the following fragment of a C program:

```
#include <stdio.h>
int main() {
    char name[64];
    ...
    printf("name: ");
    scanf("%63s",name);
    printf("Hello ");
    printf(name);
    printf("! How are you?\n");
    ...
}
```

Explain how it can be exploited to dump the content of an arbitrary memory location.

Exercise 2.2

What is a SQL injection? Write a fragment of vulnerable code, show an example of injection and then discuss how to fix your code in order to prevent SQL injections.

Exercise 2.3

CSRF is a very powerful attack that allows an intruder to inject requests into authenticated sessions. Adding random tokens inside the forms such as

```
<input type="hidden" value="Ie3eise6oiC6jeiz" name="csrf_token">
```

prevents these attacks. Explain in detail this technique and illustrate why it prevents CSRF attacks.
Optional (extra score): compare this technique with challenge-response authentication protocols.