Exercise 1

The *one-time pad* is a famous perfect cipher in which a binary plaintext string is xor-ed with a random binary key of the same length.

1. Give a formal definition of the cipher;
2. Prove that the cipher is perfect;
3. Consider a variant called *two-time pad* in which a key is reused (just once) to encrypt a second plaintext. Prove that this variant is not perfect and discuss what information is gained by any attacker that can intercept ciphertexts.

Exercise 2

We intercept a 'strange' ciphertext `0x000...000115E314E61F9` that we know to be encrypted with RSA under a key with a 2048 bit long modulus. What can we immediately conclude about the cipher public key, with very high probability, and how can we trivially compute the plaintext?

Exercise 3

Consider the following protocol

$$\begin{array}{lcll} B & \rightarrow & A & : N_B \\ A & \rightarrow & B & : E_K(B, N_B), E_K(B, K_s), N_A \\ B & \rightarrow & A & : E_{K_s}(A, N_A) \\ & \rightarrow & & : \ldots \text{secure session encrypted under } K_s \ldots \end{array}$$

where N_A and N_B are nonces, K is a long-term key shared between Alice and Bob and K_s is a fresh session key generated by Alice. The protocol aims at distributing a fresh session key between the two parties so that breaking old sessions keys never compromises future sessions. Show an attack that breaks the above property and suggest a (minimal) fix to the protocol.

Exercise 4

Consider the following candidate hash functions for 16 characters long passwords:

$$\begin{array}{rcl} h(pwd) & = & E_{pwd}(0) \\ g(pwd) & = & E_0(pwd) \end{array}$$

where E is 128-bit AES and 0 is the 128 0's bit-string.

1. Give the definition of one-way hash functions and explain how they are employed to store passwords in a secure way;
2. Discuss whether h and g above are one-way or not.

Exercise 5 (optional, gives extra score)

Define Diffie-Hellman key agreement protocol and illustrate its insecurity under active attackers.