Assignment 3

Exercise 5.1:

<u>Assume:</u> 1) the format of such payment orders is as follows: the amount in Kroner to transfer from one account to another is stored in bits 0 to 20 of the string sent, least significant bit first. 2) Assume he makes less than a million kr per month.

Exercise 5.1.1:

<u>Task:</u> What is the value of bit 20 in the payment order?

First, one needs to calculate the maximum value of 19 bits:

$$2^{19} = 524287$$

Second, the maximum value of 20 bits:

$$2^{20} = 1048576$$

Third, calculate the difference to determine the value of bit 20:

$$2^{20} - 2^{19} = 524288$$

Thus, the value of bit 20 in the payment order is 524288.

Exercise 5.1.2:

<u>Task:</u> How he can modify the encrypted payment order in such a way that he will receive more than a million kr. extra next month?

First, <u>assume</u> the security officer's salary is $\frac{1}{12}$ of the maximum value representable with 19 bits:

$$salary = \frac{524288}{12} \approx 43691$$

Second, recognize that if the security officer was to increase his monthly salary by 1 million, he would need to "change" the bits in the payment order from his monthly salary to his desired salary. Thus, first calculate his current monthly salary in binary:

$$(43691)_{10} = (00001010101010101011)_2$$

Third, convert the desired monthly salary in binary:

$$(1043691)_{10} = (11111110110011101011)_2$$

Fourth, realize that the security officer can calculate the encryption key as he knows 1) the input message, 2) the ciphertext and 3) the encryption method, namely a XOR cipher.

Fifth, calculate the key <u>assuming</u> the captured ciphertext is:

$$(00001011100010111000)_2$$

Prior to interception		
Payment order	Key	Ciphertext
$(00001010101010101011)_2$	$(00001011100010111000)_2$	$(00001011100010111000)_2$

Sixth, apply the encryption key to modify the payment order:

Post interception		
Payment order	Key	Ciphertext
$(11111110110011101011)_2$	$(00001011100010111000)_2$	$(01000110011101100000)_2$

Exercise 5.1.3:

<u>Task:</u> Is the security problem you have seen here a confidentiality problem or an authenticity problem?

This is an authenticity problem because the message has been manipulated and it does not come from where it claims to come from.

Exercise 5.1.4:

Task: The notes claim that the one-time pad cannot be broken, and yet we have identified a security problem here. Why is this not a contradiction?

In this case, it is <u>assumed</u> that the person who alters the content of the message can obtain the ciphertext and thus the encryption key by knowing 1) the bit length of the payment order and 2) the time the order is sent.

As this information is at the disposal of the adversary, it is not a bug in the algorithm, but rather a bug in the in the nature of man.

Exercise 5.1.5:

<u>Task:</u> A sender encrypts a message consisting of bits $m_1, \ldots, m_n$ with the one-time pad. Suppose an adversary intercepts the ciphertext and that he knows that m_i, the original bit at position i in the message, is 0 with probability p. The adversary wants to modify the ciphertext such that the receiver will decrypt a 0-bit at position i in the message. Show that the adversary can make the receiver obtain a 0-bit in position i with probability $max(p, 1-p)$. Optional: show that the adversary cannot do better than $max(p, 1-p)$:

If m_i is 0 with probability p then m_i is 1 with probability $1-p$, thus:

$$Pr[m_i = 0] = Pr[c_i = 0, k_i = 0] + Pr[c_i = 1, k_i = 1] = p$$

Conversely:

$$Pr[m_i = 1] = Pr[c_i = 1, k_i = 0] + Pr[c_i = 0, k_i = 1] = 1 - p$$

Since there always will be a max of 2 outcomes with 2 ways of getting to each outcome then the probability will be ½ as described on page 124 of "Secure Distributed Systems" 2021 by Ivan Damgaard, Jesper Buus Nielsen and Claudio Orlandi. The max probability will therefore never be more than $max(p, 1-p)$ due to the key being random and due to there being no other external factors which modify the probability.