ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed
Computing

FS 2016 Prof. R. Wattenhofer

Computer Engineering II

Exercise Sheet 12

We categorize questions into four different categories:

Quiz Short questions which we will solve rather interactively at the start of the exercise sessions.

Basic Improve the basic understanding of the lecture material.

Advanced Test your ability to work with the lecture content. This is the typical style of questions which appear in the exam.

Mastery Beyond the essentials, more interesting, but also more challenging. These questions are **optional**, and we do not expect you to solve such exercises during the exam.

Quiz

1 Quiz Questions

a) The CAESAR encryption works by cyclically shifting all letters by $1 \leq x \leq 25$ positions in the alphabet, e.g., a becomes d with $x = 3$. If you apply CAESAR twice with different x, is it more secure?

b) To find a large prime easily, you propose the following: You take all primes until some x, multiply them with each other, and subtract 1, resulting in a prime p. You then get another prime p' by $p' = (p+1)p - 1$, which can be iterated. Does this method work?

c) Alice wants to send a 1 if her answer is yes, and a 0 if her answer is no. To do so, she chooses a one-time pad of one bit as encryption. Is this method malleable?

d) Malleability can be prevented by hashing the ciphertext, and sending the hash together with the ciphertext.

2 Secret Sharing

a) Using some of your neighbors, replay Algorithm 12.28 with $t = 2$. Can two of you recover the chosen secret? (if you are at home, you have to do it all yourself...)

b) Why can a single participant gain no information about k?

3 The One Time Pad

a) To be extra secure, you apply the same one time pad twice to the same message. What will happen?

b) To be even more secure, you take two different one time pads and apply them to your message. Is this more secure than using a single one time pad?

c) Assume you have intercepted a message of length x, encryped with an one time pad. As x is quite short, you now just try all 2^x different keys. Explain why you still cannot decrypt the message, except knowing its length.

d) Alice and Bob agree on a one time pad of length x, upon which Alice sends Bob an encrypted message of length $2x$ using ECB (encrypting each block of length x with the OTP). Why is the message no longer encrypted with perfect secrecy? What can an eavesdropper infer from the messages?

e) Alice and Bob now heard from CBC too, and decide to use it instead of ECB. As thus, Alice picks a randomized block of length x, and then sends a message of multiple blocks to Bob, using CBC. Do we now have perfect secrecy? If not, give an example of what an attacker can infer from eavesdropping similar to with ECB above?

4 Diffie-Hellman Key Exchange

Alice and Bob want to agree on a common secret key with the Diffie-Hellman key exchange. They choose the prime $p = 7$.

a) Find all primitive roots of $p = 7$.

b) Alice picks $k_A = 4$ and Bob picks $k_B = 2$, and they agree on the smallest primitive root of $p = 7$. What is their shared common key?

c) Replay the protocol with a neighbor, using $p = 7$ and g being the biggest primitive root of p. Are you able to agree on the same key? (if you are at home, you have to do it yourself..)

d) Take the role of an eavesdropper, where Alice and Bob publicly announced $p = 5$ and $g = 3$. Alice sends 2 to Bob, and Bob sends 4 to Alice. What is their common key, when they use the Diffie-Hellman key exchange?

5 Message Authentification

a) Besides time stamps and nonces, can you think of another idea to prevent replay attacks?

b) Consider a very basic hash function h, that given some prime p, computes the hash as $h(m) = m \mod p$. Is h a collision resistant hash function? Is h a one-way hash function?

c) Assume that the discrete logarithm problem is computationally hard. Using this knowledge, come up with a one-way hash function.