

Musterlösungen zu den Hausaufgaben auf  
Blatt 12  
der Übungen zur Vorlesung  
“Grundlagen Betriebssysteme und Systemsoftware

---

G.Groh, 02.02.2009

## Aufgabe 1.1 Lösungsvorschlag

**Frage:** Was ist der Unterschied zwischen Authentifizierung und Autorisierung?

**Antwort:**

Die beiden Begriffe klingen zwar ähnlich, bezeichnen aber unterschiedliche Vorgänge:

Authentifizierung ist der Nachweis der Authentizität, also die Überprüfung der Identität des Senders.

Autorisierung meint hingegen die Zuweisung von Rechten. Jemand wird autorisiert, etwas zu tun.

Ein Zusammenhang besteht höchstens dahingehend, dass eine Person sich häufig authentifizieren muss, bevor sie autorisiert wird, etwas zu tun.

**Frage:** Was ist der Unterschied zwischen Vertraulichkeit bzw. Geheimhaltung und Integrität?

**Antwort:**

Vertraulichkeit/Geheimhaltung beschreiben den Umstand, dass es Dritten nicht möglich sein soll, Daten zu lesen.

Nur der Sender und der beabsichtigte Empfänger sollten die Nachrichteninhalte verstehen können.

Bei der Integrität geht es hingegen nicht um unerlaubtes Lesen, sondern um unerlaubtes Schreiben: Daten müssen vor nicht autorisiertem Schreiben geschützt werden.

Schreiben bedeutet dabei einfügen, modifizieren, löschen, etc.

**Frage:** Warum müssen Authentifizierung und Nachrichtenintegrität immer gemeinsam gewährleistet werden?

**Antwort:**

Es nützt nichts, wenn der Empfänger sicher sein kann, dass eine Nachricht vom wirklichen Absender stammt, wenn deren Inhalt verfälscht sein kann. Authentifizierung ohne Nachrichtenintegrität könnte dazu führen, dass ein Student eine Nachricht von der TUM bekommt, dass er eine Prüfung nicht bestanden hat. Er kann sicher sein, dass sie von der TUM ist,. Allerdings schrieb sie, dass er bestanden hat.

**Frage: RSA:** Wählen Sie  $p = 47$  und  $q = 71$ .

Berechnen Sie daraus  $n$  und  $z$ . Nehmen Sie für  $e = 79$  und  $d = 1019$  an.

**Antwort:**

- $p = 47$
- $q = 71$
- $n = 3337$
- $x = 3220$
- $e = 79$
- $d = 1019$

**Frage:** Benutzen Sie  $e$  als privaten Schlüssel und kodieren Sie damit die Nachricht "688232687966668003". Dabei müssen Sie die Nachricht in Blöcke gleicher Länge zerteilen, die jeweils kleiner als  $n$  sind.

**Antwort:**

„688“ „232“ „687“ „966“ „668“ „003“

**Frage:** Verschlüsseln.

**Antwort:** Die Verschlüsselung erfolgt gemäß:

$$c = m^e \bmod n$$

m	c
688	1570
232	2756
687	2091
966	2276
688	2423
003	158

**Frage:** Entschlüsseln.

**Antwort:** Die Entschlüsselung erfolgt analog  
gemäß:  $m = \text{mod}(c^d, n)$ .

## Aufgabe 2.1 Lösungsvorschlag

Consider a secret-key cipher that has a  $26 \times 26$  matrix with the columns headed by ABC ... Z and the rows are also ABC ... Z. Plaintext is encrypted two characters at a time. The first character is the column; the second is the row. The cell formed by the intersection of the row and column contains two ciphertext characters. What constraint must the matrix adhere to and how many keys are there?

---

The constraint is that no two cells contain the same two letters, otherwise decryption would be ambiguous. Thus each of the 676 matrix elements contains a different one of the 676 digrams. The number of different combinations is thus  $676!$ . This is a very big number.

## Aufgabe 2.2 Lösungsvorschlag

Break the following mono-alphabetic cipher (The plaintext consists of letters only and is a well known excerpt from a poem by L. Carroll)

kfd ktbd fzm eubd pzyiom mztX tu kzyg ur bzha kfthem  
ur mfudm zhX mftnm zhX mdzythe pzq ur ezsszedm zhX gthem  
zhX pfa kfd mdz tm sutythe fuk zhX pfdkfdi ntem fzld pthem  
sok pztK z stk kfd uamkdim eitdx sdruid pd fzld uoi efzk  
rui mubd ur om zid uok ur sidzKf zhX zyy ur om zid rzk  
hu foia mztX kfd ezindhkdi kfda kfzhgdx ftb boef rui kfzk

---

The result is:

the time has come the walrus said to talk of many things  
of ships and shoes and sealing wax of cabbages and kings  
of why the sea is boiling hot and whether pigs have wings  
but wait a bit the oysters cried before we have our chat  
for some of us are out of breath and all of us are fat  
no hurry said the carpenter they thanked him much for that

## Aufgabe 2.3 Lösungsvorschlag

Consider the following way to encrypt a file. The encryption algorithm uses two  $n$ -byte arrays,  $A$  and  $B$ . The first  $n$ -byte are read from the file into  $A$ . Then  $A[0]$  is copied to  $B[i]$ ,  $A[1]$  is copied to  $B[j]$ ,  $A[2]$  is copied to  $B[k]$ , etc. After all  $n$  bytes copied to the  $B$  array, that array is written to the output file and  $n$  more bytes are read into  $A$ . This procedure continues until the entire file has been encrypted. Note that here encryption is not being done by replacing characters with other ones, but by changing their order. How many keys have to be tried to exhaustively search the key space? Give an advantage of this scheme over a monoalphabetic substitution cipher?

---

The number of permutations is  $n!$ , so *this is the size of the key space*.  
One advantage is that the statistical attack based on properties of natural languages does not work because an  $E$  really does represent an  $E$ , etc.

## Aufgabe 2.4 Lösungsvorschlag

Secret key cryptography is more efficient than public key cryptography but requires the sender and receiver to agree on a secret key in advance. Suppose that the sender and receiver have never met, but there exists a trusted third party that shares a secret key with the sender and also shares a (different) secret key with the receiver. How can the sender and receiver establish a new shared secret key under these circumstances?

---

The sender picks a random key and sends it to the trusted third party encrypted with the secret key that they share. The trusted third party then decrypts the random key and reencrypts it with the secret key it shares with the receiver. This message is then sent to the receiver.

## Aufgabe 2.5 Lösungsvorschlag

Not having the computer echo the password is safer than having it echo an asterisk for each character since the latter discloses the password length to anyone who can see the screen. Assuming that passwords consist of upper- and lower-case letters and digits only, and that passwords are between 5 and 8 characters long: How much safer is it not displaying anything (as in Unix)?

---

It depends on how long the password is. The alphabet from which passwords is built has 62 symbols. The total search space is  $62^5 + 62^6 + 62^7 + 62^8$ , which is about  $2 \times 10^{14}$ . If the password is known to be  $k$  characters, the search space is reduced to only  $62^k$ . The ratio of these is thus  $2 \times 10^{14} / 62^k$ . For  $k$  from 5 to 8, these values are 242,235, 3907, 63, and 1. In other words, learning that the password is only 5 characters reduces the search space by a factor of 242,235 because all the long passwords do not have to be tried. This is a big win. However, learning that it is eight characters does not help much because it means that all the short (easy) passwords can be skipped.