2 Shift Ciphers

Assume that the alphabet is linearly ordered. A shift cipher replaces each letter of the plaintext by the letter that follows a certain number k of positions in the alphabet. If the end of the alphabet is reached, restart at the beginning. That means, we consider cyclic shifts. The number k is the key.

Decryption works in the reverse direction: Count backwards from the ciphertext letter.

Example 1: Original Caesar

Here $\Sigma = \{\text{A,...,Z}\} = \mathcal{Z}_{26}$, hence $A \leftrightarrow 0, B \leftrightarrow 1, ..., Z \leftrightarrow 25$. Caesar used the fixed key $k = 3$. Encryption looks like follows

```
C A E S A R  | +3  (plaintext)
-----------
F D H V D U        (ciphertext)
```

> Note that the original Roman alphabet had only 23 letters without J, U, W. However in this part of the lecture we (almost) always use the 26 letter alphabet.

As key space we could also take $K = \mathbb{Z}$. Then the key length is ∞. But effectively we only have 26 different encryption functions, one of them being trivial. Therefore the effective key length is only $\log_2(26) \approx 4.7$.

Example 2: Rot13

ROT13 is a shift cipher over the alphabet {A, ..., Z} that shifts each letter by 13 positions ahead in the alphabet. As mnemonic take the table

```
A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z
```

As encryption function this is almost useless. Its purpose is hiding some texts, say of offensive content, from immediate recognition. The reader of the message can figure it out only by a conscious act.

Because $13 + 13 = 26$, double encryption restores the plaintext. That is, ROT13 is an involution. Or in other words: encryption = decryption as functions.

Example 3: XOR

This example extends the notion of shift cipher towards the more general version given in the mathematical description below. In this sense XOR is a shift cipher on the space of l-bit blocks. Thus our alphabet is the l-dimensional vector space $\mathbb{F}_2^l$ over the two element field $\mathbb{F}_2$. The operation

XOR is the addition of vectors in this space (because XOR of bits is the addition in the field $\mathbb{F}_2$). The key is a fixed block k. Each plaintext block a is XORed with k bitwise, that is, "shifted" (or translated) by k.

Mathematical Description

Let the alphabet Σ be a finite group G with n elements and with group composition $*$. As key space also take $K = G$. For $k \in K$ let

$$f_k : \Sigma^* \longrightarrow \Sigma^*$$

be the continuation of the right translation $f_k(s) = s * k$ for $s \in \Sigma$, that is

$$f_k(a_1, \ldots, a_r) = (a_1 * k, \ldots, a_r * k) \quad \text{for } a = (a_1, \ldots, a_r) \in \Sigma^r.$$

The effective key length is $d(F) = \log_2(n)$. Thus the key space is quite small and is easily completely searched except when n is VERY LARGE. An example will follow in the next section.