Chapter 2
Stream Ciphers

If we look at the types of cryptographic algorithms that exist in a little bit more detail, we see that the symmetric ciphers can be divided into stream ciphers and block ciphers, as shown in Fig. 2.1.

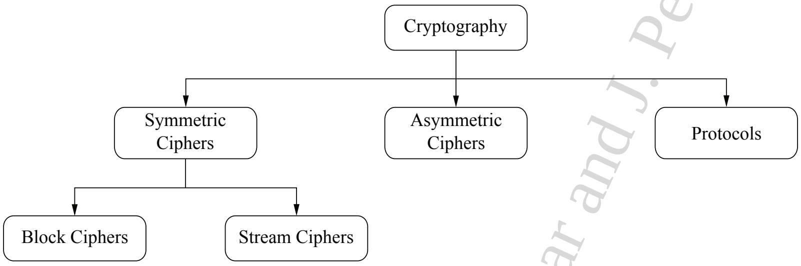

Fig. 2.1 Main areas within cryptography

This chapter gives an introduction to stream ciphers:

- The pros and cons of stream ciphers
- Random and pseudorandom number generators
- A truly unbreakable cipher: the One-Time Pad (OTP)
- Linear feedback shift registers and Trivium, a modern stream cipher

2.1 Introduction

2.1.1 Stream Ciphers vs. Block Ciphers

Symmetric cryptography is split into block ciphers and stream ciphers, which are easy to distinguish. Figure 2.2 depicts the operational differences between stream (Fig. 2.2a) and block (Fig. 2.2b) ciphers when we want to encrypt *b* bits at a time, where *b* is the width of the block cipher.

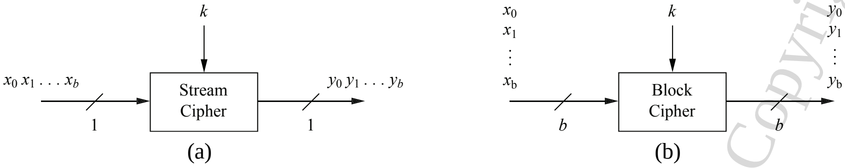

Fig. 2.2 Principles of encrypting *b* bits with a stream (a) and a block (b) cipher

A description of the principles of the two types of symmetric ciphers follows.

Stream ciphers encrypt bits individually. This is achieved by adding a bit from a *key stream* to a plaintext bit. There are synchronous stream ciphers where the key stream depends only on the key, and asynchronous ones where the key stream also depends on the ciphertext. If the dotted line in Fig. 2.3 is present, the stream cipher is an asynchronous one. Most practical stream ciphers are synchronous ones and Sect. 2.3 of this chapter will deal with them. An example of an asynchronous stream cipher is the cipher feedback (CFB) mode introduced in Sect. 5.1.4.

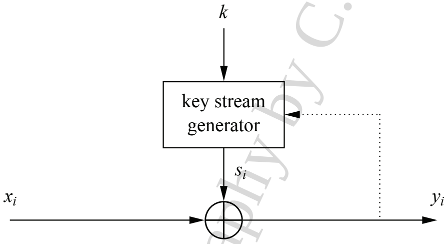

Fig. 2.3 Synchronous and asynchronous stream ciphers

Block ciphers encrypt an entire block of plaintext bits at a time with the same key. This means that the encryption of any plaintext bit in a given block depends on every other plaintext bit in the same block. In practice, the vast majority of block ciphers either have a block length of 128 bits (16 bytes) such as the advanced encryption standard (AES), or a block length of 64 bits (8 bytes) such as the data encryption standard (DES) or triple DES (3DES) algorithm. All of these ciphers are introduced in later chapters.

This chapter gives an introduction to stream ciphers. Before we go into more detail, it will be helpful to learn some useful facts about stream ciphers vs. block ciphers:

1. In practice, in particular for encrypting computer communication on the Internet, block ciphers are used more often than stream ciphers.
2. Because stream ciphers tend to be small and fast, they are particularly relevant for applications with little computational resources, e.g., for cell phones or other small embedded devices. A prominent example for a stream cipher is the A5/1 cipher, which is part of the GSM mobile phone standard and is used for voice encryption. However, stream ciphers are sometimes also used for encrypting Internet traffic, especially the stream cipher RC4.
3. Traditionally, it was assumed that stream ciphers tended to encrypt more efficiently than block ciphers. *Efficient* for software-optimized stream ciphers means that they need fewer processor instructions (or processor cycles) to encrypt one bit of plaintext. For hardware-optimized stream ciphers, *efficient* means they need fewer gates (or smaller chip area) than a block cipher for encrypting at the same data rate. However, modern block ciphers such as AES are also very efficient in software. Moreover, for hardware, there are also highly efficient block ciphers, such as PRESENT, which are as efficient as very compact stream ciphers.

2.1.2 Encryption and Decryption with Stream Ciphers

As mentioned above, stream ciphers encrypt plaintext bits individually. The question now is: How does encryption of an individual bit work? The answer is surprisingly simple: Each bit x_i is encrypted by adding a secret key stream bit s_i modulo 2.

Definition 2.1.1 Stream Cipher Encryption and Decryption
The plaintext, the ciphertext and the key stream consist of individual bits,
i.e., $x_i, y_i, s_i \in \{0,1\}$.
Encryption: $y_i = e_{s_i}(x_i) \equiv x_i + s_i \bmod 2$.
Decryption: $x_i = d_{s_i}(y_i) \equiv y_i + s_i \bmod 2$.

Since encryption and decryption functions are both simple additions modulo 2, we can depict the basic operation of a stream cipher as shown in Fig. 2.4. Note that we use a circle with an addition sign as the symbol for modulo 2 addition.

Just looking at the formulae, there are three points about the stream cipher encryption and decryption function which we should clarify:

1. Encryption and decryption are the same functions!