Block Ciphers — A Survey

Lars R. Knudsen

K.U.Leuven, Dept. Elektrotechniek–ESAT
Kard. Mercierlaan 94, B-3001 Heverlee,
lars.knudsen@esat.kuleuven.ac.be

1 Introduction

In this paper we give a short overview of the state of the art of secret key block ciphers. We focus on the main application of block ciphers, namely for encryption. The most important known attacks on block ciphers are linear cryptanalysis and differential cryptanalysis. Linear cryptanalysis makes use of so-called *linear hulls* i.e., the parity of a subset of plaintext bits which after a certain number of rounds equals the parity of a subset of ciphertext bits with a probability sufficiently far away from one half. Differential cryptanalysis makes use of so-called *differentials* (A, B), i.e., a pair of plaintexts with difference A, which after a certain number of rounds result in a difference B with a high probability. The hulls and differentials can be used to derive (parts of) the secret key.

Also, several extensions of the two above attacks have been introduced lately: the truncated differential attack [38,39], the higher order differential attack [43,38,28], the multiple linear attack [30], and the non-linear/linear attack [41]. Also, a combination of the two methods, the differential-linear attack [27], has been considered. Other (general) attacks are the non-surjective attack [68] and the interpolation attack [28].

To improve resistance against differential and linear cryptanalysis it has been suggested to use power polynomials in a finite field [3,62,51]. On the other hand, it has been shown that if a cipher consists solely of such functions other efficient attacks become possible [28]. Another well-known way of improving the security of a block cipher is by means of multiple encryption, i.e., where a plaintext block is processed several times using the same (component) block cipher with different keys.

In § 2 an introduction to block ciphers is given and § 3 lists and discusses the modes of operation for encryption. In § 4 we describe the theoretical and practical security of block ciphers. The most important methods of cryptanalysing block ciphers are given in § 5. § 6 discusses design principles of block ciphers, in particular it is shown how to build ciphers immune to the attacks described in previous sections. The theory of multiple encryption is described in § 7. In § 8 we summarise our results.

2 Block Ciphers - Introduction

The history of cryptography is long and goes back at least 4000 years to the Egyptians, who used hieroglyphic codes for inscription on tombs [18]. Since then

B. Preneel, V. Rijmen (Eds.): COSIC'97 Course, LNCS 1528, pp. 18–48, 1998.

many cryptosystems, also called ciphers, have been developed and used. Many of these old ciphers are much too weak to be used in applications today, because of the tremendous progress in computer technology. There are essentially two types of cryptosystems, one-key and two-key ciphers. In one-key ciphers the encryption of a plaintext and the decryption of the corresponding ciphertext is performed using the same key. Until 1976 when Diffie and Hellman introduced *public-key* or two-key cryptography [21] all ciphers were one-key systems. Therefore one-key ciphers are also called conventional cryptosystems. Conventional cryptosystems are widely used throughout the world today, and new systems are published frequently. There are two kinds of one-key ciphers, stream ciphers and block ciphers. In stream ciphers a long sequence of bits is generated from a short string of key bits, and is then added bitwise modulo 2 to the plaintext to produce the ciphertext. In block ciphers the plaintext is divided into blocks of a fixed length, which are then encrypted into blocks of ciphertexts using the same key. Block ciphers can be divided into three groups: substitution ciphers, transposition ciphers and product ciphers. In the following a few examples of the different types of block ciphers are given.

Notation: Let $\mathcal{A_M}$ and $\mathcal{A_C}$ be the alphabets for plaintexts and ciphertexts, respectively. Let $M = m_0, m_1, \ldots, m_{n-1}$ be an n-character plaintext, s.t. for every i, $m_i \in \mathcal{A_M}$ and let $C = c_0, c_1, \ldots, c_{n-1}$ be a ciphertext, s.t. for every i, $c_i \in \mathcal{A_C}$. We assume that an alphabet $\mathcal{A_X}$ is isomorphic with $\mathbb{N}_{\mathcal{A_X}}$.

2.1 Substitution Ciphers

As indicated in the name every plaintext character is substituted by some ciphertext character. There are four kinds of substitution ciphers.

- Simple substitution
- Polyalphabetic substitution
- Homophonic substitution
- Polygram substitution

We restrict ourselves to consider substitution ciphers of the first two kinds.

Simple Substitution In a cipher with a simple substitution each plaintext character is transformed into a ciphertext character via the same function **f**. More formally, $\forall i \; : \; 0 \leq i < n$

$$\begin{aligned} \mathbf{f} \; &: \; \mathcal{A_M} \rightarrow \mathcal{A_C} \\ c_i &= \mathbf{f}(m_i) \end{aligned}$$

It is believed that Julius Caesar encrypted messages by shifting every letter in the plaintext 3 positions to the right in the alphabet. This cipher is based on *shifted alphabets*, i.e., $\mathcal{A_M} = \mathcal{A_C}$, and is in general defined as follows $\mathbf{f}(m_i) = m_i + k \pmod{|\mathcal{A_M}|}$. For the Caesar cipher the secret key k is the number 3. In general, the cipher is easily broken in at most $|\mathcal{A_M}|$ trials. Shift the ciphertexts one position until the plaintext arises.

Polyalphabetic Substitution In a polyalphabetic substitution the plaintext characters are transformed into ciphertext characters using a j-character key $K = k_0, \ldots, k_{j-1}$, which defines j distinct functions $\mathbf{f}_{k_0}, \ldots, \mathbf{f}_{k_{j-1}}$. More formally $\forall i \ : \ 0 < i \leq n$

$$\begin{aligned} \mathbf{f}_{k_l} &: \mathcal{A}_\mathcal{M} \rightarrow \mathcal{A}_\mathcal{C} \quad \forall l \ : \ 0 \leq l < j \\ c_i &= \mathbf{f}_{k_{i \bmod j}}(m_i) \end{aligned}$$

The Vigenère cipher was first published in 1586 [19]. Let us assume again that $\mathcal{A}_\mathcal{M} = \mathcal{A}_\mathcal{C}$. Then the Vigenère cipher is defined as follows

$$c_i = \mathbf{f}_{k_{i \bmod j}}(m_i) = m_i + k_{i \bmod j} \pmod{|\mathcal{A}_\mathcal{M}|}$$

2.2 Transposition Systems

Transposition systems are essentially permutations of the plaintext characters. Therefore $\mathcal{A}_\mathcal{M} = \mathcal{A}_\mathcal{C}$. A transposition cipher is defined as follows $\forall i \ : \ 0 \leq i < n$

$$\begin{aligned} \mathbf{f} &: \mathcal{A}_\mathcal{M} \rightarrow \mathcal{A}_\mathcal{M} \\ \eta &: \{0, \ldots, (n-1)\} \rightarrow \{0, \ldots, (n-1)\}, \text{ a permutation} \\ c_i &= \mathbf{f}(m_i) = m_{\eta(i)} \end{aligned}$$

Many transposition ciphers permute characters with a fixed period j. In that case

$$\begin{aligned} \mathbf{f} &: \mathcal{A}_\mathcal{M} \rightarrow \mathcal{A}_\mathcal{M} \\ \eta &: \{0, \ldots, (j-1)\} \rightarrow \{0, \ldots, (j-1)\}, \text{ a permutation} \\ c_i &= \mathbf{f}(m_i) = m_{(i \operatorname{div} j) + \eta(i \bmod j)} \end{aligned}$$

The Vigenère and in general substitution ciphers can be broken when enough ciphertext is available to the cryptanalyst by the index of coincidence, Kasiski's method, etc. [18,19,29]. Transposition ciphers can be broken using the frequency distributions for digrams, trigrams and N-grams [18,19,29]. The interested reader will find a comprehensive treatment of early cryptanalysis in [29].

2.3 Product Systems

An obvious attempt to make stronger ciphers than the ones we have seen so far, is to combine substitution and transposition ciphers. These ciphers are called *product ciphers.* Many product ciphers have been developed, including Rotor machines [18]. Most of the block ciphers in use today are product ciphers. A product cipher is called an *iterated cipher* if the ciphertext is computed by iteratively applying a round function several times to the plaintext. In each round a round key is combined with the text input.

Definition 1. *In an r-***round iterated block cipher** *the ciphertext is computed by iteratively applying a round function g to the plaintext, s.t.*

$$C_i = g(C_{i-1}, K_i), \;\; i = 1, ...r, \tag{1}$$

where C_0 is the plaintext, K_i a round key and C_r is the ciphertext. Decryption is done by reversing (1), therefore, for a fixed key K_i, g must be invertible.

In this paper we consider only iterated block ciphers and assume that the plaintexts and ciphertexts are bit strings of equal length.

A special class of iterated ciphers are the *Feistel* ciphers, named after Horst Feistel [23]

Definition 2. *A* **Feistel cipher** *with block size $2n$ and with r rounds is defined as follows. The round function is defined*

$$g : GF(2^n) \times GF(2^n) \times GF(2^m) \rightarrow GF(2^n) \times GF(2^n)$$

$$g(X, Y, Z) = (Y, \; F(Y, Z) + X)$$

where g can be any function taking two arguments of n bits and m bits respectively and producing n bits. '+' is a commutative group operation on the set of n-bit blocks. Given a plaintext $P = (P^L, P^R)$ and r round keys $K_1, K_2, ..., K_r$ the ciphertext $C = (C^L, C^R)$ is computed in r rounds. Set $C_0^L = P^L$ and $C_0^R = P^R$ and compute for $i = 1, 2, ..., r$

$$(C_i^L, C_i^R) = (C_{i-1}^R, F(C_{i-1}^R, K_i) + C_{i-1}^L)$$

Set $C_i = (C_i^L, C_i^R)$ and $C^L = C_r^R$ and $C^R = C_r^L$. The round keys $(K_1, ..., K_r)$, where $K_i \in GF(2^m)$, are computed by a key schedule algorithm on input a master key K.

We will assume that '+' is the bitwise exclusive-or operation, if not explicitly stated otherwise.

The Data Encryption Standard (DES) [63] is by far the most widely used iterated block cipher today. Around the world, governments, banks, and standards organisations have made the DES the basis of secure and authentic communication [75]. The DES is a Feistel cipher, but with special properties. In general we define the so-called DES-like iterated ciphers.

Definition 3. *A* **DES-like iterated cipher** *is a Feistel cipher, where the F function is defined*

$$\begin{aligned} F(X, K_i) &= f(E(X) + K_i) \\ f &: GF(2^m) \rightarrow GF(2^n), \;\; m \geq n \\ E &: GF(2^n) \rightarrow GF(2^m), \text{ an affine expansion mapping} \end{aligned}$$

3 Modes of Operations

The most obvious and widespread use of a block cipher is for encryption. In 1980 a list of four modes of operation for the DES was published [64]. These four modes can be used with any block cipher and seem to cover most applications of block ciphers used for encryption [18]. In the following let $E_K(\cdot)$ be the permutation induced by using the block cipher E of block length n with the key K and let $P_1, P_2, ..., P_i, ...$ be the blocks of plaintexts to be encrypted. The four modes are

- **Electronic Code Book (ECB)** The native mode, where one block at a time is encrypted independently of the encryptions of other blocks.

$$\begin{aligned} \text{Encryption:} \quad C_i &= E_K(P_i) \\ \text{Decryption:} \quad P_i &= E_K(C_i) \end{aligned}$$

- **Cipher Block Chaining (CBC)** The chaining mode, where the encryption of a block depends on the encryptions of previous blocks.

$$\text{Encryption:} \quad C_i = E_K(P_i \oplus C_{i-1})$$

$$\text{Decryption:} \quad P_i = D_K(C_i) \oplus C_{i-1}$$

  where C_0 is a chosen initial value.
- **Cipher Feedback (CFB)** The first stream mode, where one m-bit character at a time is encrypted.

$$\begin{aligned} \text{Encryption:} \quad C_i &= P_i \oplus \text{MSB}_m(E_K(X_i)) \\ X_{i+1} &= \text{LSB}_{n-m}(X_i) \parallel C_i \end{aligned}$$

$$\begin{aligned} \text{Decryption:} \quad P_i &= C_i \oplus \text{MSB}_m(E_K(X_i)) \\ X_{i+1} &= \text{LSB}_{n-m}(X_i) \parallel C_i \end{aligned}$$

  where X_1 is a chosen initial value, $\parallel$ denotes concatenation of blocks, MSB_s and LSB_s denote the s most and least significant bits respectively or equivalently the leftmost and rightmost bits respectively. Here m can be any number between 1 and the block length of the cipher. If the plaintext consists of characters $m = 7$ or $m = 8$ is usually the well-chosen parameter.
- **Output Feedback (OFB)** The second stream mode, where the stream bits are not dependent on the previous plaintexts, i.e., only the stream bits are fed back, not the ciphertext as in CFB mode.

$$\begin{aligned} \text{Encryption:} \quad C_i &= P_i \oplus \text{MSB}_m(E_K(X_i)) \\ X_{i+1} &= \text{LSB}_{n-m}(X_i) \parallel \text{MSB}_m(E_K(X_i)) \end{aligned}$$

$$\begin{aligned} \text{Decryption:} \quad P_i &= C_i \oplus \text{MSB}_m(E_K(X_i)) \\ X_{i+1} &= \text{LSB}_{n-m}(X_i) \parallel \text{MSB}_m(E_K(X_i)) \end{aligned}$$

  where X_1 is a chosen initial value.

In fact, both the CFB and OFB modes have two parameters, the size of the plaintext block and the size of the feedback value. In the above definition we have chosen them equal and will do so also in the following.

The ECB is the native mode, well-suited for encryption of keys of fixed length. It is not suited for the encryption of larger plaintexts, since equal blocks are encrypted into equal blocks. To avoid this, the CBC mode is recommended. Not only does a current ciphertext block depend on the current plaintext but also on all previous ciphertext blocks. In some applications there is a need for encryptions of characters, instead of whole blocks, e.g., 8 bytes for the CBC mode of DES. For that purpose the CFB and OFB modes are suitable. It is often recommended to use the OFB mode only with full feedback, i.e., with $m = n$ (64 for the DES). It comes from the fact, that for $m < n$ the feedback function is not one-to-one, and therefore has a relatively short cycle [18] of length about $2^{n/2}$. Furthermore the initial value X_1 in the OFB mode should be chosen uniformly at random. For example, in the case where X_1 is the concatenation of n/m equal m-bit blocks, say $(a \,\|\, a \,\|\, ... \,\|\, a)$, for about 2^{k-m} keys $\mathrm{MSB}_m(E_K(X_1)) = a$. Therefore $X_2 = X_1$ and in general $X_i = X_1$. This is not dangerous for the CFB mode, where the X_i's are also dependent on the plaintext.

Error Propagation An important issue in the applications of the four modes is how an error in the transmission of ciphertexts is propagated. In the ECB mode an error in a ciphertext block affects only one plaintext block. A lost ciphertext block results in a lost plaintext block. An error in a ciphertext block in the CBC mode affects two plaintexts blocks. As an example, assume that ciphertext C_3 has an error and that all other ciphertext blocks are error-free, then $P_4 = D_K(C_4) \oplus C_3$ inherits the error from C_3 and $P_3 = D_K(C_3) \oplus C_2$ will be completely garbled. Here we assume that even a small change in the input to the block cipher will produce a randomly looking output. All other plaintexts will be decrypted correctly. A lost ciphertext block results in a lost plaintext block and an error in one addition plaintext block. The mode synchronises itself after that. In the CFB mode an error in a ciphertext block C_i will be inherited by the corresponding plaintext block P_i, and moreover since X_{i+1} contains the garbled C_i the subsequent plaintexts blocks will be garbled until the X value is free of C_i, i.e., when C_i has been shifted out. In other words in CFB mode with m-bit ciphertexts, at most $n/m + 1$ plaintext blocks will be garbled. The case of lost ciphertext blocks is similar to that of the CBC mode. In the OFB mode, since the feedback is independent of the plain- and ciphertexts, a transmission error in a ciphertext block garbles only the corresponding plaintext block and is not propagated to other plaintext blocks. On the other hand, a lost ciphertext block will result in an infinite error propagation.

4 Security of Secret Key Block Ciphers

When discussing the security of cryptographic systems one needs to define a model of the reality. We will use the model of Shannon [73], which is depicted in Figure 1.

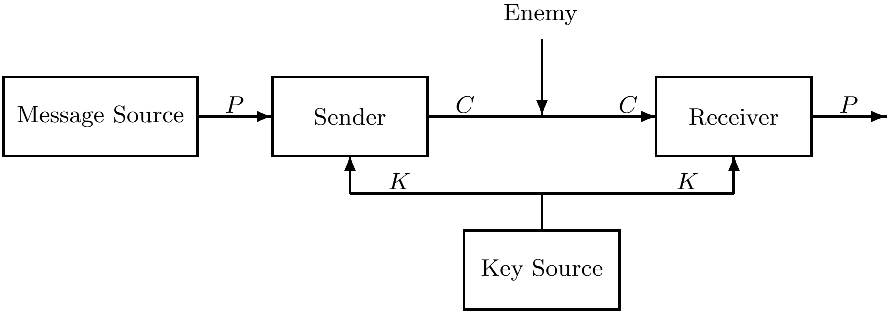

Fig. 1. Shannon's model of a general secrecy system.

The sender and the receiver share a common key K, which has been transmitted over a secure channel. The sender encrypts a plaintext P using the secret key K, sends C over an insecure channel to the receiver, who restores C into P using K. The attacker has access to the insecure channel and can intercept the ciphertexts (cryptograms) sent from the sender to the receiver. In this section we assume that the legitimate sender and receiver use a secret key cipher $E_K(\cdot)$ of block size n (bits), where the key K is of size k bits. To avoid an attacker to speculate in how the legitimate parties have constructed their common key, we will assume

Assumption 1 *All keys are equally likely and a key K is always chosen uniformly random.*

Also we will assume that all details about the cryptographic algorithm used by the sender and receiver are known to the attacker, except for the secret key. This assumption is known as Kerckhoffs's Assumption [29].

Assumption 2 *The enemy cryptanalyst knows all details of the enciphering process and deciphering process except for the value of the secret key.*

4.1 Classification of Attacks

Using these assumptions we classify the possible attacks an attacker can do.

- **Ciphertext only attack.** The attacker possesses a set of intercepted ciphertexts.

- **Known plaintext attack.** The attacker obtains $P_1, P_2, ..., P_s$ a set of s plaintexts and the corresponding ciphertexts $C_1, C_2, ..., C_s$.
- **Chosen plaintext attack.** The attacker chooses *a priori* a set of s plaintexts $P_1, P_2, ..., P_s$ and obtains in some way the corresponding ciphertexts $C_1, C_2, ..., C_s$.
- **Adaptively chosen plaintext attack.** The attacker chooses a set of plaintexts $P_1, P_2, ..., P_s$ interactively as he obtains the corresponding ciphertexts $C_1, C_2, ..., C_s$. That is, the attacker chooses P_1, obtains C_1, **then** chooses P_2 etc.
- **Chosen ciphertext attacks.** For symmetric ciphers these are similar to those of chosen plaintext attack and adaptively chosen plaintext attack, where the roles of plain- and ciphertexts are interchanged.

The chosen text attacks are obviously the most powerful attacks. In many applications they are however also unrealistic attacks. If the plaintext space contains redundancy, it will be hard for an attacker to 'trick' a legitimate sender into encrypting non-meaningful plaintexts and similarly hard to get ciphertexts decrypted, which do not yield meaningful plaintexts. But if a system is secure against an adaptively chosen plaintext/ciphertext attack then it is also secure against all other attacks. An ideal situation for a designer would be to prove that her system is secure against an adaptively chosen plaintext attack, although an attacker may never be able to mount more than a ciphertext only attack.

4.2 Theoretical Secrecy

In his milestone paper from 1949 [73] Shannon defines perfect secrecy for secret key systems and shows that they exist. We will now give a brief description of Shannon's theory and the most important results. Let **P**, **C** and **K** be the random variables representing the plaintexts, ciphertexts and the keys respectively. Let $P_{\mathbf{X}}(x)$ be the probability that the random variable **X** takes on the value x.

Definition 4 ([73]). *The uncertainty (entropy) $H(\mathbf{X})$ of a random variable $\mathbf{X}$ is defined as the expectation of the negative logarithm of the corresponding probability distribution.*

Using the logarithm base 2, we get

$$H(\mathbf{X}) \overset{\text{def}}{=} E[-log_2 P_{\mathbf{X}}(x)] = - \sum_{x \in supp(P_{\mathbf{X}})} P_{\mathbf{X}}(x) \times log_2 P_{\mathbf{X}}(x)$$

where $supp(P_{\mathbf{X}}) \overset{\text{def}}{=} \{x : P_{\mathbf{X}}(x) \neq 0\}$. When using this logarithm the entropy of **X** can be seen as the number of bits needed to represent (the possible values of) **X** in an optimal binary coded form. The **conditional entropy of X given Y** is defined as

$$\begin{aligned} H(\mathbf{X}|\mathbf{Y}) &\overset{\text{def}}{=} E[-log_2 P_{X|Y}(X|Y)] \\ &= - \sum_{x,y \in supp(P_{\mathbf{X},\mathbf{Y}})} P_{\mathbf{X},\mathbf{Y}}(x,y) \times log_2 P_{\mathbf{X}|\mathbf{Y}}(x|y). \end{aligned}$$

In other words the uncertainty about $\mathbf{X}$ given that $\mathbf{Y}$ is known. The quantity $I(X;Y) = H(\mathbf{X}) - H(\mathbf{X}|\mathbf{Y})$ is called **the information** that $\mathbf{Y}$ gives about $\mathbf{X}$.

Definition 5 ([73]). *A secret key cipher is* **perfect** *if and only if $H(\mathbf{P}) = H(\mathbf{P}|\mathbf{C})$, i.e., when the ciphertext $\mathbf{C}$ gives no information about the plaintext $\mathbf{P}$.*

This definition leads to the following result.

Corollary 1. *A perfect cipher is unconditionally secure against a ciphertext only attack.*

As noted by Shannon the Vernam cipher, also called the *one-time pad*, is a perfect secret key cipher. In the one-time pad the plaintext characters are exclusive-or'ed with independent key characters to produce the ciphertexts. However, the practical applications of perfect secret key ciphers are limited, since, as also noted by Shannon, it requires as many digits of secret key as there are digits to be enciphered [46]. A less stringent form of theoretical secrecy is possible, defined by Shannon in terms of

Definition 6 ([73]). *The* **unicity distance***, n_{ud}, of a cipher is the smallest number s such that there is essentially only one value of the secret key K that is consistent with the ciphertexts $C_1, ..., C_s$.*

In other words, the unicity distance is the smallest s, s.t.

$$H(K|C_1, ..., C_s) \simeq 0$$

The unicity distance depends on both the key size and on the redundancy in the plaintext space. Redundancy is an effect of the fact that certain plaintext characters appear more frequently than others. For a block cipher of size n, **the redundancy** ρ is defined as

$$\rho = 1 - H(\mathbf{P})/n$$

where $\mathbf{P}$ is the random variable representing the plaintexts. $H(\mathbf{P})/n$ estimates the average number of bits of information per bit in a plaintext.

Theorem 1 ([73]). *The unicity distance of a cipher is*

$$n_{ud} = \frac{H(\mathbf{K})}{\rho}$$

where ρ is the redundancy of the plaintext space.

The smallest number N_{ud}, such that N_{ud} is a multiple of the block size n and $N_{ud} \geq n_{ud}$, is the least number of ciphertext bits an attacker needs from a legitimate sender in order to at least in principle be able to determine a unique key in a ciphertext only attack.

Example 1 ([46]). The redundancy of English language messages is about 0.8. So for the DES, $k = 56$, $n = 64$ and

$$n_{ud} = \frac{56}{0.8} \simeq 70$$

Therefore N_{ud} is 128 bits, the same as two ciphertext blocks.

Although the unicity distance is small as in the example, it does not necessarily mean that the DES can be broken using only 2 known ciphertexts. First of all, Shannon's measures are made using a random cipher model, but more important, the unicity distance gives no indication of the computational difficulty in breaking a cipher, merely a lower bound on the amount of ciphertext needed in a ciphertext only attack. However, if the plaintext space contains (close to) no redundancy, the unicity distance will tend to infinity, i.e., $n_{ud} \leadsto \infty$ as $\rho \leadsto 0$. In this case a ciphertext only attack will never succeed. A cipher, for which it can be shown that $H(K|C_1, ..., C_s)$ never approaches zero, even for large s, is called a **strongly ideal** cipher.

One way to remove the redundancy in a plaintext space is by data compression, but no known methods achieve perfect data compression [42]. Since perfect and strongly ideal ciphers are both impractical, we also consider computationally secrecy, or practical secrecy.

4.3 Practical Secrecy

Traditionally, cryptanalysis has been very focused on finding the key K of a secret key cipher. However, there are other serious attacks, which do not find the secret key. We classify the types of breaking a block cipher as follows, inspired by the classification of forgeries on digital signature systems given by Goldwasser, Micali and Rivest in [24,25].

- **Total break.** An attacker finds the secret key K.
- **Global deduction.** An attacker finds an algorithm A, functionally equivalent to $E_K(\cdot)$ (or $D_K(\cdot)$) without knowing the key K.
- **Instance (local) deduction.** An attacker finds the plaintext (ciphertext) of an intercepted ciphertext (plaintext), which he did not obtain from the legitimate sender.
- **Information deduction.** An attacker gains some (Shannon) information about key, plaintexts or ciphertexts, which he did not get directly from the sender and which he did not have before the attack.

Clearly, this classification is hierarchical, i.e., if a total break is possible, then a global deduction is possible etc. We assume that all the above attacks are independent of how the keys used by the legitimate parties are chosen, i.e., we use Assumption 1. A global deduction is possible when a block cipher contains a "block structure". If certain subsets of the ciphertext are independent of certain subsets of the plaintext, then no matter how long the key is, the block cipher is vulnerable to a global deduction in a known plaintext attack. Also, in iterated block ciphers the round keys are sometimes generated in a one-way fashion [69,72,16,17]. So in attacks, which find the round keys, it may be impossible for the attacker to derive the actual value of the secret key, but at the same time the round keys enable the attacker to encrypt and decrypt. An instance deduction can be as dangerous as a total break, if the number of likely plaintexts is small. Consider the situation where the block cipher is used for encrypting a key in a key-exchange protocol. Here only one plaintext is encrypted and a total break is equal to an instance deduction. Information deduction is the least serious attack, however the legitimate parties are often interested in that no information at all about the plaintexts are obtained by any enemies, which is particularly dangerous if the plaintext space is highly redundant.

Brute-Force (Trivial) Attacks.

- **Total break.** All block ciphers are totally breakable in a ciphertext only attack, simply by trying all keys one by one and checking whether the computed plaintext is meaningful, using only about N_{ud} ciphertexts. This attack requires the computation of about 2^k encryptions.
  Alternatively, one has the table look-up attack, where the attacker, encrypts in a pre-computation phase a fixed plaintext P under all possible keys and sorts and stores all the ciphertexts. Thereafter the cipher is total breakable in a chosen plaintext attack requiring one chosen plaintext. There might be some keys encrypting P into the same ciphertext. Repeating the attack a few times with $P' \neq P$ will give a unique key.
- **Global/instance deduction.** All block ciphers are globally/instance deducible under a known/chosen plaintext attack. Simply get and store all possible plaintext/ciphertext pairs. The running time of a deduction is the time of one table look-up.
- **Information deduction.** All block ciphers are information deducible in a ciphertext only attack. Consider a block cipher used in the ECB mode. Denote two plaintexts by P_i and P_j and assume that an attacker intercepted the two corresponding ciphertext blocks, C_i and C_j. It follows that $H(P_i, P_j | C_i, C_j) < H(P_i, P_j)$, since $C_i \neq C_j \Rightarrow P_i \neq P_j$, and $C_i = C_j \Rightarrow P_i = P_j$[1]. Since $I(P_i, P_j; C_i, C_j) = H(P_i, P_j) - H(P_i, P_j | C_i, C_j)$, it follows that $I(P_i, P_j; C_i, C_j) > 0$, i.e., the attacker gains information about the plaintext blocks from two ciphertext blocks. Obviously, the more ciphertext blocks available to the attacker the more information is gained. A similar result holds for other modes.

The information deduction just shown is trivial and results in only small information. The following general result shows that a non-trivial information gain can be obtained when about the square root of all ciphertexts are available.

Theorem 2 ([35]). *Every n-bit block cipher used in the ECB, CBC or CFB mode is information deducible in a ciphertext only attack with complexity about* $2^{n/2}$.

Note that the result of Theorem 2 is independent of the key size.

Also, Hellman [26] has presented a time-memory trade-off attack on any block cipher, which finds the secret key after $2^{2k/3}$ encryptions using $2^{2k/3}$ words of

[1] Here we assume, that the attacker does not a priori have this information about the plaintexts.

memory. The $2^{2k/3}$ words of memory are computed in a pre-processing phase, which takes the time of 2^k encryptions.

To estimate the complexity of a cryptanalytic attack one must consider both the time it takes, the amount of data that is needed and the storage requirements. For an n-bit block cipher the following complexities should be considered.

- **Data complexity.** The amount of data needed as input to an attack. Units are measured in blocks of length n. We denote this complexity C_d.
- **Processing complexity.** The time needed to perform an attack. Time units are measured as the number of encryptions an attacker has to do himself. We denote this complexity C_p.
- **Storage complexity.** The words of memory needed to do the attack. Units are measured in blocks of length n. We denote this complexity C_s.

As a rule of thumb, the complexity of an attack is taken to be the maximum of the three complexities, i.e., $C_a = max(C_d, C_p, C_s)$. In general, there are some deviations from this rule and furthermore the three types of complexity of an attack are relative to the attacker. As an example, we may say that the above attack by Hellman [26] on the DES has complexity $2^{2\times 56/3} \simeq 2^{38}$. Although the time of the pre-computation phase is 2^{56}, first of all, it is done only once after which any DES-key can be derived with complexity 2^{38}, secondly 2^{56} DES encryptions can be done reasonable fast in hardware on specially designed machines [81]. On the other hand, the storage requirements may be unrealistic for most attackers, e.g., the attack on the DES will require about 1000 Gigabytes of memory.

5 Cryptanalysis of Block Ciphers

The history of cryptanalysis is long and at least as fascinating as the history of cryptography. As a single example, in 1917 in an article in "Scientific American" the Vigenère cipher was claimed to be "impossible of translation" [19]. Today, it is an exercise in cryptography classes to illustrate that this claim is not true.

5.1 Differential Cryptanalysis

The most well-known method of analysing conventional cryptosystems today is *differential cryptanalysis*, published by Eli Biham and Adi Shamir in 1990. The method has proved to be very efficient and cryptosystems, which have been conjectured strong, have been broken, for some systems (e.g., GDES) almost alarmingly easy [5]. Differential cryptanalysis has been applied to a wide range of iterated ciphers including the DES [63], GDES [70,71], Lucifer [76,2], FEAL [74,56], LOKI'89 [9,34], REDOC [12], PES [44], Khafre [54], SAFER [47,48,37], RC5 [69], and IDEA [44,8]. For this reason the differential attack must be considered one of the most general cryptanalytic attacks known to date. Furthermore, differential cryptanalysis has caused the revision and redesign of several cryptosystems and was the first attack which could (theoretically) recover DES keys in time less than the expected cost of exhaustive search [5,6]. Differential cryptanalysis is universal in that it can be used against any cryptographic mapping which is constructed from iterating a fixed round function. We will give a brief description of differential cryptanalysis with respect to a general n-bit iterated cipher, cf., Definition 1.

One usually defines a **difference** between two bit strings, X and X' of equal length as

$$\Delta X = X \otimes (X')^{-1}, \tag{2}$$

where $\otimes$ is the group operation on the group of bit strings used to combine the key with the text input in the round function and where $(X)^{-1}$ is the inverse element of X with respect to $\otimes$. The idea behind this is, that the differences between the texts before and after the key is combined are equal, i.e., the difference is independent of the key. To see this, note that

$$(X \otimes K) \otimes (X' \otimes K)^{-1} = X \otimes K \otimes K^{-1} \otimes X'^{-1} = X \otimes (X')^{-1} = \Delta X.$$

However, in a strong encryption algorithm there will be some components which are non-linear in the $\otimes$-operation. In a differential attack one exploits that for certain input differences the distribution of output differences of the non-linear components is non-uniform.

Definition 7. *An s-round* characteristic *is a series of differences defined as an $s+1$-tuple $\{\alpha_0, \alpha_1, \ldots, \alpha_s\}$, where $\Delta P = \alpha_0$, $\Delta C_i = \alpha_i$ for $1 \leq i \leq s$.*

The probability of a characteristic is derived from the probability that ΔC_i is the difference after i rounds given that ΔC_{i-1} is the difference after $i-1$ rounds. Define p_i as the probability that inputs of difference α_{i-1} lead to output of difference α_i, where the probability is taken over all choices of the round key and the inputs to the ith round. In [44] the notion of a Markov cipher was introduced. In a Markov cipher this probability is independent of the actual inputs of the round and is calculated over all possible choices of the round key. Also in [44] it was shown that in a Markov cipher if the round keys K_i are independent then the p_i's are also independent and

$$\Pr(\Delta C_s = \alpha_s \,|\, \Delta P_0 = \alpha_0) = \prod_{i=1}^{s} \Pr(\Delta C_i = \alpha_i \,|\, \Delta C_{i-1} = \alpha_{i-1}). \tag{3}$$

Experimental results on DES, LOKI, and FEAL [5,35] have shown that in these ciphers (3) is a good approximation of the probability, when the round keys are dependent, e.g., derived from a key schedule algorithm. Although in general one defines a difference according to (2), for some ciphers such as RC5, it "pays off" to choose another difference, which was illustrated in [40].

Assume without loss of generality that the operation $\otimes$ is is the exclusive-or operation ($\oplus$). Consider an iterated block cipher as defined in Definition 1. Let C_r and C'_r be the ciphertexts for some plaintext pair. In a chosen plaintext attack the cryptanalyst does not know the inputs C_{r-1} and C'_{r-1} to the final round. However, a characteristic can be chosen so that the difference of the ciphertexts after $r-1$ rounds of encryptions, ΔC_{r-1}, is known either completely or partially with probability p. Then for two ciphertexts C, C' the cryptanalyst will try to solve the following equation for K_r

$$g^{-1}(C_r, K_r) \oplus g^{-1}(C'_r, K_r) = \Delta C_{r-1}. \tag{4}$$

Sometimes one does not recover the entire value of K_r, and the remaining key bits are then found by an exhaustive search. The method of differential cryptanalysis can be summarized as follows:

1. Find a highly probable $(r-1)$-round characteristic $\{\Delta P, \Delta C_1, \ldots, \Delta C_{r-1}\}$ which (partially) predicts ΔC_{r-1}.
2. Select a random plaintext P, compute P' according to ΔP and get the encryptions of the pair. Determine candidate round keys k_r, which satisfy (4). Increment a counter for each candidate round key k_r.
3. Repeat Step 2 until one round key k_r is distinguished as being counted significantly more often than other round keys. Take k_r to be the actual round key K_r.

In some differential attacks using an $(r-1)$-round characteristic only the plaintext difference ΔP and the last ciphertext difference ΔC_{r-1} need to be fixed. That is, the intermediate differences $\Delta C_1, \Delta C_2, \ldots, \Delta C_{r-2}$ can have any value. Lai and Massey introduced the notion of *differentials* [44].

Definition 8. *An s-round* differential *is a pair of differences $\{\alpha_0, \alpha_s\}$, where $\Delta P = \alpha_0$, $\Delta C_s = \alpha_s$.*

The probability of an s-round differential $(\Delta P, \Delta C_s)$ is the conditional probability that given an input difference ΔP at the first round, the output difference at the sth round will be ΔC_s. More formally, the probability of an s-round differential is given as

$$\Pr(\Delta C_s = \beta_s \mid \Delta P = \beta_0) = \sum_{\beta_1} \cdots \sum_{\beta_{s-1}} \prod_{i=1}^{s} \Pr(\Delta C_i = \beta_i \mid \Delta C_{i-1} = \beta_{i-1}), \tag{5}$$

where $\Delta C_0 = \Delta P$. A differential will, in general, have a higher probability than a corresponding characteristic. Differentials were used by Knudsen and Nyberg to show how to obtain immunity against differential attacks [62]. Also, for some ciphers there is a significant advantage in considering differentials instead of characteristics. As an example, the differential used to attack RC5 in [40] with $w = 32$ and 12 rounds has a probability of 2^{-53} and a corresponding characteristic has a probability of only 2^{-96}.

Experiments on restricted versions of DES [5] showed that the number of chosen plaintexts needed by the differential attack is approximately $1/p$, where p is the probability of the differential being used.

In a differential attack the attacker does not know the key. Therefore in finding a good differential, the attacker computes the probabilities of differentials assuming that all the round keys are uniformly random and independent. However, the pairs of encryption an attacker gets are encrypted using the same key, where the round keys are fixed and (can be) dependent. Put informally "there is a difference between what an attacker can expect to see and what he actually sees". In [42] this problem is dealt with as follows

Definition 9 ((Hypothesis of stochastic equivalence)). *For virtually all high probability $(r-1)$-round differentials (α, β)*

$$Pr_P(\Delta C_1 \ = \ \beta \mid \Delta P \ = \ \alpha, \ K = k) \ \approx \ Pr_{P,K}(\Delta C_1 \ = \ \beta \mid \Delta P \ = \ \alpha,)$$

holds for a substantial fraction of the key values k.

In a recent differential attack by Knudsen and Rijmen on IDEA [8], it was exploited that the hypothesis of stochastic equivalence does not hold for IDEA reduced to 3,5 rounds.

Higher Order Differentials In [43] Lai gave a definition of higher order derivatives of discrete functions. Later Knudsen used higher order differentials to cryptanalyse ciphers presumably secure against conventional differential attacks, i.e., attacks based on first order differentials [38]. Jakobsen and Knudsen [28] extended these attacks and applied them to the cipher of [62]. We refer to [43,38] for the definitions of higher order differentials. By the *reduced cipher*, we denote the cipher that one gets by removing the final round of the original cipher.

The output bits of the reduced cipher are all expressible as polynomials $GF(2)[x_1, x_2, \ldots, x_n]$, where $x_1, x_2, \ldots, x_n$ are (a subset of) plaintext bits. Assume that these polynomials have degree not higher than d. Then according to [43, Proposition 2] (see also [38]), we have

$$\sum_{x \in \mathcal{L}_d} p(x) = c, \tag{6}$$

where $\mathcal{L}_d$ denotes a d-dimensional subspace of $GF(2)^n$, c is the same for any space parallel to $\mathcal{L}_d$, and p is a function which computes the output from the reduced cipher. It follows that

$$\sigma(w) = \sum_{x \in \mathcal{L}_{d+1}} p(x+w) = 0 \text{ for all } w \in GF(2)^n \tag{7}$$

if and only if $p(x)$ is a polynomial of degree d or lower. If d is sufficiently low, the block cipher can be attacked as follows. For all values of the key in the last round, decrypt all ciphertexts one round, obtaining the output of the reduced cipher, and compute the value of $\sigma(w)$. The keys for which $\sigma(w)$ ends up being zero are candidates for the correct value of the last-round key. By repeating the attack a few times only one (or a few) values of the last-round key will be left suggested. Jakobsen and Knudsen applied this method to the cipher example given in [62]. This cipher is "provably secure" against a differential attack. In a higher order differential attack this cipher is broken using only $2^9 = 512$ chosen plaintexts and a running time of 2^{41}.

Truncated Differentials In some ciphers it is possible and advantageous to predict the values of parts of the differences after each round of the cipher. The notion of truncated differentials was introduced by Knudsen in [38]:

Definition 10. *A differential that predicts only parts of an n-bit value is called a* truncated differential*. More formally, let (a, b) be an i-round differential. If a' is a subsequence of a and b' is a subsequence of b, then (a', b') is called an i-round truncated differential.*

A truncated differential can be seen as a collection of differentials. As an example, consider an n-bit block cipher and the truncated differential (a', b), where a' specifies the least $n' < n$ significant bits of the plaintext difference and b specifies the ciphertext difference of length n. This differential is a collection of all $2^{n-n'}$ differentials (a, b), where a is any value, which truncated to the n' least significant bits is a'.

The truncated differentials were used in [39] to attack 5 rounds of the 6 round SAFER K [47,48] in time only 2^{37} with 2^{45} chosen plaintexts. In [48] a differential attack using conventional differentials on SAFER K with 5 rounds was estimated to require more computations than a brute-force exhaustive attack. Also, in [8] a truncated differential attack was presented on 3,5 rounds of IDEA [44].

5.2 Linear Cryptanalysis

Linear cryptanalysis was proposed by Matsui in 1993 [49]. A preliminary version of the attack on FEAL was described in 1992 [52]. Linear cryptanalysis [49] is a known plaintext attack in which the attacker exploits linear approximations of some bits of the plaintext, ciphertext and key. In the attack on the DES (or on DES-like iterated ciphers) the linear approximations are obtained by combining approximations for each round under the assumption of independent round keys. The attacker hopes in this way to find an expression (8), which holds with probability $p_L \neq \frac{1}{2}$ over all keys [49], such that $|p_L - \frac{1}{2}|$, called the bias, is maximal.

$$(P \cdot \alpha) \oplus (C \cdot \beta) = (K \cdot \gamma) \tag{8}$$

where $P, C, \alpha, \beta, \gamma$ are m-bit strings and where '$\cdot$' denotes the dot product.

Given an approximation (8) a linear attack using N plaintexts and the N corresponding ciphertexts goes as follows.

Linear attack [49]

1. For all plaintexts, P, and ciphertexts, C, let T be the number of times the left hand side of (8) is zero.

2. If $T > N/2$ guess that $K \cdot \gamma = 0$, otherwise guess that $K \cdot \gamma = 1$.

The attack finds one bit of information about the key, $K \cdot \gamma$, and the complexity of a successful attack, i.e., the number of known plaintexts needed, using the above algorithm can be approximated in the following way. Let $\mathbf{T}$ be a binomial random variable taking on the value 0 with probability p. Assume that $|p-1/2|$ is small and without loss of generality that $p > 1/2$. Then

$$\Pr(\mathbf{T} > N/2) = \Phi(2\sqrt{N}|p-1/2|),$$

where Φ is the normal distribution function. With $N = |p-1/2|^{-2}$ the success rate is about 97.72%. Since the required number of plaintexts is the dominating factor in a linear attack, the complexity, N_P, of the above linear attack is [49]

$$N_P \simeq |p_L - 1/2|^{-2}$$

where p_L is the probability of a linear approximation of the form (8). This estimate shows that the quantity of interest in a linear attack is $|p_L - 1/2|^{-2}$, the reciprocal of the square of the bias. For DES-like iterated ciphers linear approximations of the form (8) can be found by combining linear approximations of each round in the cipher. As in differential cryptanalysis one can define characteristics to be used in linear cryptanalysis [49].

The above linear attack is not very efficient, since it finds only one bit of information about the key. However, there exists an extended linear attack, which finds more key bits. For the DES the linear approximations used by Matsui affects at most one S-box per round. Only six key bits affect affect an S-box directly, so instead of approximating the first and last round one can simply repeat the attack for all values of the relevant key bits in those two rounds. One gets the following approximation

$$(P \cdot \alpha) \oplus (C \cdot \beta) \oplus (F(P_R, K_1) \cdot \alpha_1) \oplus (F(C_R, K_r) \cdot \alpha_r) = (K \cdot \gamma) \qquad (9)$$

where P_R, C_R are the right halves of the plain- and ciphertexts respectively. K_1 and K_r are the key bits affecting the linear approximation in the first and rth rounds. For all choices of the keys K_1 and K_r the approximation (9) can be seen as an approximation of a cipher of $r-2$ rounds, i.e., two rounds shorter than the original cipher. The attack goes as follows with N available plaintexts.

Extended linear attack [49]

1. For all, say n, values of the two keys, K_1 and K_r do:
For all plaintexts, P, and ciphertexts, C, let T_i, $i = 1, ..., n$, be the number of times the left hand side of (9) is zero.

2. Let T_{max} and T_{min} be the maximum and minimum values of the T_i's for $i = 1, ..., n$. If $|T_{max} - N/2| > |T_{min} - N/2|$ guess that K_1 and K_r are the key values from the computation of T_{max}.
If $|T_{max} - N/2| < |T_{min} - N/2|$ guess that K_1 and K_r are the key values from the computation of T_{min}.

In case of the DES it is conjectured and confirmed by computer experiments [49,50] that the efficiency of (9) decreases, when the values of K_1 or K_r are incorrect values. In [49,50] it is estimated that the complexity of an extended linear attack on the DES with up to 16 rounds is about

$$N_P \simeq c \times |p_L - 1/2|^{-2}$$

where $c \leq 8$ [50]. Note that the practicality of this extended attack depends also on how many key bits are needed to count on in the first and last rounds.

In [30] Kaliski and Robshaw showed an improved linear attack using multiple linear approximations. In [32] Knudsen and Robshaw showed a linear attack using non-linear approximations in the outer rounds of an iterated cipher. Both these attacks have not yet been shown to offer a significant improvement in attacks on the DES compared to Matsui's linear attack. The attacks seem best suited for attacks on ciphers with large S-boxes, such as LOKI [9,32].

Similar to the concept of differentials in differential cryptanalysis is the concept of *linear hulls* in linear cryptanalysis introduced in [60], based on the following generalisation of Parseval's Theorem. Let $X \in GF(2)^m$ and $K \in GF(2)^\ell$ be random variables and $Y = Y(X, K)$, $Y \in GF(2)^n$, be a random variable which is a function of X and K.

Theorem 3. *If X and K are independent and K is uniformly distributed, then for all $a \in GF(2)^m$, $b \in GF(2)^n$ and $\gamma \in GF(2)^\ell$*

$$2^{-\ell} \sum_{k \in GF(2)^\ell} | P_X(X \cdot a + Y(X,k) \cdot b = 0) - 1/2 |^2 =$$

$$2^{-\ell} \sum_{k \in GF(2)^\ell} | P_X(X \cdot a + Y(X,k) \cdot b + k \cdot \gamma = 0) - 1/2 |^2 =$$

$$\sum_{c \in GF(2)^\ell} | P_{X,K}(X \cdot a + Y(X,K) \cdot b + K \cdot c = 0) - 1/2 |^2$$

This theorem says that the probability of an approximation (8) does not depend on the value of γ. Moreover for the probability p of a linear approximation it holds that $|p - 1/2|^2$ is the sum of $|p_\gamma - 1/2|^2$ for all values of γ.

5.3 Differential-Linear Attack

In [27] Hellman and Langford showed how to combine the techniques of differential and linear attacks. The attack is a chosen plaintext attack and considers pairs of plaintexts and ciphertexts, the bits of which are (partly) approximated by linear approximations. In particular, they illustrated the attack by devising an attack of the DES reduced to 8 rounds, which on input only 512 chosen plaintexts finds the secret key. It seems that the attack is not easily extended to more than 8 rounds of DES [27].

In [1] Aoki and Ohta applied the differential-linear attack to FEAL. The attack takes a long time, but only 12 chosen plaintexts are needed.

5.4 Interpolation Attack

In [28] Jakobsen and Knudsen introduced a new attack on block ciphers. The attack is based on the following well-known formula. Let R be a field. Given $2n$ elements $x_1, \ldots, x_n$, $y_1, \ldots, y_n \in R$, where the x_is are distinct. Define

$$f(x) = \sum_{i=1}^{n} y_i \prod_{1 \leq j \leq n, j \neq i} \frac{x - x_j}{x_i - x_j}. \tag{10}$$

$f(x)$ is the only polynomial over R of degree at most $n-1$ such that $f(x_i) = y_i$ for $i = 1, \ldots, n$. Equation (10) is known as the *Lagrange interpolation formula* (see e.g., [10, page 185]).

In the *interpolation attack* an attacker constructs polynomials using pairs of plaintexts and ciphertexts. This is particularly easy if the components in the cipher can be easily expressed as mathematical functions. The idea in the attack is, that if the constructed polynomials have a small degree, only few plaintexts and their corresponding ciphertexts are necessary to solve for the (key-dependent) coefficients of the polynomial, e.g., using Lagrange's interpolation. To recover key bits one expresses the ciphertext before the last round as a polynomial of the plaintext. Subsequently, for every value of (parts of) the last-round key one decrypts all ciphertexts one round and uses these values in the Lagrange interpolation. If a few extra plaintexts and ciphertexts fit into this polynomial, the correct value of the key is found with a high probability. The attack can be repeated until only one value of the last-round key is suggested. In an extended version of the attack meet-in-middle techniques are used to further reduce the degrees of the used polynomials [28]. In particular, Jakobsen and Knudsen showed how to attack ciphers, which are provably secure against differential and linear attacks.

5.5 Key Schedule Attacks

In this section we consider the key schedules of block ciphers. Much research on the DES has been focused on the S-boxes, but a weak key schedule can be exploited in cryptanalytic attacks.

We consider an n-bit block cipher, where $E_K(\cdot)$ denotes encryption with the key K and $D_K(\cdot)$ denotes decryption.

Definition 11. *A weak key K, is a key for which encryption equals decryption, i.e., $E_K(X) = D_K(X)$ for all n-bit texts X.*

Definition 12. *A pair of semi-weak keys K, K^*, are keys for which encryption with one keys equals decryption with the other key, i.e., $E_K(X) = D_{K^*}(X)$ for all n-bit texts X or equivalently, $D_K(X) = E_{K^*}(X)$ for all n-bit texts X.*

It is well-known that there are at least four weak keys and six pairs of semi-weak keys for the DES. In [11] D. Coppersmith showed that there are exactly 2^{32} fixed points for the DES used with a weak key.

If there are only a small number of weak keys they pose no problem for applications of encryption if the used keys are chosen uniformly at random. However, when block ciphers are used in hash modes where e.g., the key input can be chosen by the attacker in attempts to find collisions, they play an important role as demonstrated in [15,65].

In [13] Daemen showed that there exist a large class of 2^{51} easy-identifiable keys for IDEA. These keys can be identified using only few plaintexts and ciphertexts. Note that IDEA uses 128-bit keys. In [80] Vaudenay showed that for 1 in 2^{15} keys for Blowfish a differential attack is faster than an exhaustive key search. In [40] Knudsen and Meier showed that there exist a large class of differentially weak keys for RC5 [69], keys for which a specific differential attack has improved performance.

Related Key Attacks In this section we consider the *related key* attacks. There are several variants of this attack depending on how powerful the attacker is assumed to be.

1. Attacker gets encryptions under one key.
2. Attacker gets encryptions under several keys.
   (a) Known relation between keys.
   (b) Chosen relation between keys.

Knudsen introduced the method by giving a chosen plaintext attack of the first kind on LOKI'91 [33], reducing an exhaustive key search by almost a factor of four. Later Biham improved the attack [4] on LOKI'91, reducing an exhaustive key search by almost a factor of six, and also introduced the second kind of related key attacks. Still later Knudsen described a related key attack on SAFER K [37] and recently, Kelsey, Schneier, and Wagner [31] applied the related key attacks to a wide range of block ciphers.

Note that for the attacks of 2b above we have to omit Assumption 1. It may be argued that the attacks with a chosen relation between the keys are unrealistic. The attacker need to get encryptions under several keys, in some attacks even with chosen plaintexts. However there exist quite realistic settings, in which an attacker may succeed to obtain such encryptions, as argued in [31]. Also, there exists quite efficient methods to preclude the related key attacks [31,17].

6 Design of Block Ciphers

In this section we discuss some of the problems involved in the design of a block cipher.

6.1 Design Principles

Two generally accepted design principles for practical ciphers are the principles of confusion and diffusion that were suggested by Shannon. In his own words: *"The method of confusion is to make the relation between the simple statistics of the ciphertext and the simple description of the key a very complex and involved one". "In the method of diffusion the statistical structure of the plaintext which leads to its redundancy is dissipated into long range statistics"* [73]. Massey[46] interprets Shannon's concepts of confusion and diffusion as follows

> **Confusion**
> *The ciphertext statistics should depend on the plaintext statistics in a manner too complicated to be exploited by the cryptanalyst.*
>
> **Diffusion**
> *Each digit of the plaintext and each digit of the secret key should influence many digits of the ciphertext.*

Shannon's design principles are very general and informal. There have been many suggestions in the past of how to obtain good properties (diffusion/confusion) for a block cipher, but so far there is no known exact recipe of how to construct a secure block cipher. A necessary and obvious requirement is that the cipher is resistant against all known attacks, e.g., differential and linear attacks.

We stress that a cryptographic design principle should not be over valued. Design principles should be seen as "guidelines" in the construction of ciphers, evolved from years of experience, and as necessary, but **not** sufficient requirements. There are many examples of this in the history of cryptography. We already mentioned the example of [28], where a block cipher "provably secure" against differential and linear attacks was broken by some other means. Also, in [45] the group properties of a cryptosystem based on permutation groups were studied. It was claimed that the ability of a system to generate the symmetric group on the message space is "one of the strongest security conditions that can be offered". In [57] an example of a cipher was given, that generates the symmetric group, but still is a weak cipher vulnerable to a known plaintext attack.

6.2 Block and Key Size

It is clear from the discussion in Section 4.1 that if either the block or key size is too small or both, a block cipher is vulnerable to a brute force attack. These attacks are independent of the internal structure and intrinsic properties of an algorithm. Most block ciphers, e.g., DES, IDEA, FEAL, LOKI, SAFER, and SHARK have a block size of 64 bits. For these ciphers the birthday attack of Theorem 2 require storage/collection of 2^{32} ciphertext blocks for a success of about one half. This may not seem to be a realistic attack. First of all, it seems unlikely that a single key is used to process that many ciphertexts, second storage of 2^{32} ciphertext blocks of each 64 bits will require about 2^5 Gigabytes of memory. Still, if an attacker can predict approximately how frequently a key is changed, he can repeat his attack several times with fewer ciphertext blocks and get a high probability of success. This should be taken into consideration, when designing new block ciphers.

The key size of the DES is only 56 bits, which is too short. In [81] a design of an exhaustive search machine was given, which at the cost of 1 million US$ finds the secret key of the DES in average time 3.5 hours. Most of the newest block cipher proposals have a larger key, e.g., IDEA [42], SAFER SK-128 [37,47], SHARK [67], and SQUARE [14] have key sizes of 128 bits. RC5 [69] and Blowfish

[72] have variable key lengths. In [7] is was estimated that with respect to an exhaustive key search a key size of at least 90 bits will suffice for the next 20 years.

6.3 Resistance Against Differential Attacks

We consider an r-round iterated block cipher with round function g. Denote by p_g the highest probability of a non-trivial one-round differential achievable by the cryptanalyst.

$$p_g = \max_{\beta} \max_{\alpha \neq 0} \ \Pr_K(\Delta C_1 = \beta \mid \Delta P = \alpha) \tag{11}$$

where the probabilities are taken over all possible keys. In the following we will omit the subscript of the probabilities. The probability of a differential is given by (5). It is easy to obtain a lower bound of any differential in an r-round iterated cipher expressed in terms of p_g.

Theorem 4 ([35]). *Consider an r-round iterated cipher, which has independent round keys. Any s-round differential, $s \geq 1$, has a probability of at most p_g, where p_g is the probability of the most likely one-round differential.*

For DES-like iterated ciphers, Theorem 4 is trivial, since $p_g = 1$, when the right halves of a pair of inputs are equal. For DES-like iterated ciphers, these differentials are called trivial one-round differentials. It is possible to get a lower bound on all differentials in a DES-like iterated cipher expressed in terms of the most likely non-trivial one-round differential. Let now p_{max} denote

$$p_{max} = \max_{\beta} \max_{\alpha_R \neq 0} \ \Pr(\Delta C_1 = \beta \mid \Delta P = \alpha) \tag{12}$$

where α_R is the right half of α. We assume in the following that $p_{max} < 1$.

Theorem 5 ([62]). *Consider an r-round iterated DES-like cipher with independent round keys. Any s-round differential, $s \geq 4$, has a probability of at most $2p_{max}^2$.*

In the following section it is shown that the round function in an iterated cipher can be chosen in such a way that the probability of any non-trivial one-round differential, p_{max}, is small.

6.4 Resistance Against Linear Attacks

As in differential cryptanalysis it is possible to get a lower bound on all linear approximations of an iterated cipher expressed in terms of the most likely one-round approximation. Let p be the probability of a linear approximation. Then $|p-1/2|$ is called the bias. Recall that the success of a linear attack is proportional to the reciprocal value of the square of the bias of the used linear approximation. Matsui showed how to treat differential and linear cryptanalysis in a similar way [51] by defining $q = (2p-1)^2$. Let now q_{max} denote the highest such quantity for a one-round linear approximation. Then the following result holds.

$f(x)$	p_{max}	q_{max}	$ord(f)$	Conditions
x^{2^k+1}	2^{s-n}		2	$s = gcd(k, n)$
x^{2^k+1}		2^{s-n}		$s = gcd(k, n)$, $\frac{n}{s}$ odd
$(x^{2^k+1})^{-1}$	2^{1-n}	2^{1-n}	$(n + 1)/2$	$gcd(k, n) = 1$, n odd
x^{-1}	2^{1-n}	2^{2-n}	$n - 1$	n odd
x^{-1}	2^{2-n}	2^{2-n}	$n - 1$	n even

Table 1. Mappings in $GF(2^n)$.

Theorem 6 ([62,51]). *Consider an r-round iterated DES-like cipher with independent round keys. Any s-round linear hull, $s \geq 4$, has a reciprocal squared bias of at most $2q_{max}^2$.*

In the following we show that there exist functions for which q_{max} of every non-trivial one-round linear hull is small.

Let $N(f)$ denote the non-linearity of f, i.e., the smallest of the Hamming distances of any non-zero linear combination of the output coordinate functions to the set of all affine functions [59]. For a function f, where $f : GF(2^n) \rightarrow GF(2^m)$ any linear approximation of f is bounded as follows,

$$q_{max} \leq (2\frac{2^{m-1} - N(f)}{2^m})^2 = (1 - \frac{N(f)}{2^{m-1}})^2.$$

Differentially Uniform, Nonlinear Mappings By using the functions studied in [61,3,58,20] one can obtain round functions in a DES-like cipher such that p_{max} and q_{max} are small. We summarise the results of [61,3,58] in Table 1, where $ord(f)$ is the order of the coordinate functions of f. Note that squaring in $GF(2^n)$ over $GF(2)$ is a linear function, which means that for any of functions $f(x) = x^d$ in Table 1 it holds for the functions $g(x) = (f(x))^{2^l} = x^{d2^l}$ that $p^f_{max} = p^g_{max}$ and $q^f_{max} = q^g_{max}$. Using these mappings and Theorems 5 and 6 it is possible to construct block ciphers, for which one can show that every s-round differential or linear hull has a very low probability.

6.5 Resistance Against Other Attacks

As mentioned earlier one should be careful not to focus too much on the resistance against a limited set of attacks, when constructing new block ciphers. In some cases other attacks become possible. E.g., for some of the mappings shown above the output bits are only quadratic functions of the input bits, thereby enabling higher order differential attacks.

Let E be a n-bit r-round iterated block cipher. Assume that the nonlinear order of the ciphertext bits after one round is d and d^s after s rounds with a high probability. Then higher order differential attacks will in general not be possible after r rounds, if $d^r \simeq n$. One should take into account that the attacker may be able to guess key bits in the outer rounds of the cipher thereby attacking a cipher with a fewer number of rounds. Thus, if the nonlinear order should reach the block size after, say, $r-2$ rounds.

It is yet unknown how to obtain exact security against truncated differential attacks. However, a truncated differential is a collection of differentials. Therefore, if the probabilities of all differentials can be sufficiently lower-bounded, this attack will have only small probability of succeeding.

The differential-linear attack will only work if both good linear hulls and good differentials exist. Thus, the techniques of the previous section also apply in this case.

The interpolation attack works particularly well when the outputs of one round of E can be described as a polynomial of the input bits with relatively few nonzero coefficients. Thus, if (some elements of) E cannot be described as such, it seems that the attack will not be possible. But the interpolation attack is a very new approach and needs further study.

The key-schedule attacks can be precluded by using only so-called strong key-schedules [36], see also [31,17].

7 Cascade Ciphers

In this section, we look at methods for enhancing cryptosystems based on the idea of encrypting plaintext blocks more than once. In a *cascade of ciphers* it is assumed that the keys of the component ciphers are independent. The following result was proved by Even and Goldreich.

Theorem 7 ([22]). *A cascade of ciphers is at least as hard to break as any of the component ciphers in attacks where an attacker cannot make use of plaintext statistics.*

As seen, the result establishes a connection between the security of a cascade of ciphers and of the underlying ciphers. The following result covering all attacks was proved by Maurer and Massey.

Theorem 8 ([53]). *Under any attack, a cascade of ciphers is at least as hard to break as the first cipher.*

The two results hold for any reasonable definition of breaking a cipher [22,53], e.g., they hold for key-recovery attacks as well as for attacks that find a plaintext given a ciphertext.

7.1 Multiple Encryption

A special case of a cascade of ciphers is when the component ciphers are equal, also called multiple encryption. In the following we consider different forms of multiple encryption. In the following let $\mathcal{X}$ the underlying encryption scheme, and let E_K and D_K denote encryption and decryption respectively, in $\mathcal{X}$ under key K. We assume that the key space of $\mathcal{X}$ consists of all k-bit strings and that the block length of $\mathcal{X}$ is m.

Double Encryption The simplest idea one could think of would be to encrypt twice using two keys K_1, K_2, i.e., let the ciphertext corresponding to P be $C = E_{K_2}(E_{K_1}(P))$. It is clear (and well-known), however, that no matter how K_1, K_2 are generated, there is a simple meet-in-the middle attack that breaks this system in a known plaintext attack using 2^k encryptions and 2^k blocks of memory, i.e., the same time complexity as key search in the original system. The memory requirements can be reduced significantly by using the methods of Wiener and van Oorschot [79], and it is clear that this is not a satisfactory improvement over $\mathcal{X}$.

Triple Encryption Triple encryption with three independent keys K_1, K_2, and K_3, where the ciphertext corresponding to P is $C = E_{K_3}(E_{K_2}(E_{K_1}(P)))$, is also not a satisfactory solution for a similar reason as for double encryption. A simple meet-in-the-middle attack will break this in time about 2^{2k} encryptions and space 2^k blocks of memory. Thus we do not get full return for our effort in tripling the key length. We would like attacks to take time close to 2^{3k}, if the key length is $3k$. In addition to this, if $\mathcal{X} = DES$, then a simple triple encryption would preserve the complementation property, and preserve the existence of weak keys. Recently, it was shown that if an attacker can mount a related key attack, triple encryption can be broken in time about 2^k [31]. The attack requires that the attacker can get the encryptions of a small number of known plaintexts under two sets of keys. The two triples of keys must differ only in the third keys with a difference known to the attacker.

It is clear, however, that no matter how the three keys in triple encryption are generated, the meet-in-the-middle attack mentioned is still possible, and so the time complexity of the best attack against *any* triple encryption variant is not larger than 2^{2k}. It therefore seems reasonable to try to generate the three keys from two independent $\mathcal{X}$-keys K_1, K_2, since triple encryption will not provide security equivalent to more than 2 keys anyway.

Two-Key Triple Encryption One variant of this idea is well-known as two-key triple encryption, proposed by W. Tuchmann [77]: we let the ciphertext corresponding to P be $E_{K_1}(D_{K_2}(E_{K_1}(P)))$. Compatibility with a single encryption can be obtained by setting $K_1 = K_2$. As one can see, this scheme uses a particular, very simple way of generating the three keys from K_1, K_2.

Theorem 9 ([17]). *In attacks where an attacker cannot make use of plaintext statistics two-key triple encryption is at least as hard to break as it is to break a cryptosystem that uses a single decryption function of the underlying block cipher for encryption.*

Even though this result establishes some connection between the security of two-key triple encryption with the underlying cipher, it does not (seem to) hold for any attacks.

It is interesting to note that the related-key attack on a triple encryption scheme is not applicable to two-key triple encryption [31]. However each of K_1

and K_2 influences only particular parts of the encryption process. Because of this, variants of the meet-in-the-middle attack are possible that are even faster than exhaustive search for K_1, K_2. In [55] Merkle and Hellman describes an attack on two-key triple DES encryption requiring 2^{56} chosen plaintext-ciphertext pairs and a running time of 2^{56} encryptions using 2^{56} words of memory. This attack was refined in [78] into a known plaintext attack on the DES, which on input n plaintext-ciphertext pairs finds the secret key in time $2^{120}/n$ using n words of memory. The attacks can be applied to any block cipher.

In [17] stronger methods for generating the keys is given. The main idea is to generate them *pseudorandomly* from 2 $\mathcal{X}$ keys using a generator based on the security of $\mathcal{X}$. In this way, an enemy trying to break $\mathcal{Y}$ either has to treat the 3 keys as if they were really random which means he has to break $\mathcal{X}$, according to Theorem 8; or he has to use the dependency between the keys — this mean breaking the generator which was also based on $\mathcal{X}$. As a concrete example the **3-PEK** scheme (for triple encryption with pseudorandomly expanded keys) was proposed. As before, the key length of $\mathcal{X}$ is k and the block length is m. First, the three keys X_1, X_2, X_3 are generated:

$$\begin{aligned} X_1 &= E_{K1}(E_{K2}(IV_1)) \\ X_2 &= E_{K1}(E_{K2}(IV_2)) \\ X_3 &= E_{K1}(E_{K2}(IV_3)) \end{aligned}$$

where IV_i are three different initial values, e.g. $IV_i = C + i$, where C is a constant. Subsequently, the three keys X_i are used as keys for triple encryption. It is shown in [17] that if $\mathcal{X}$ is secure then so is $\mathcal{Y}$ and at the same time, the meet-in-the-middle attacks of [55,78] and the related key attack on triple encryption [31] are not possible. Using DES as the underlying cipher, 3-PEK has the additional advantage to other schemes, that there are no weak keys and that the complementation property does not hold.

8 Conclusion

This paper surveyed the state of the art of cryptanalysis of block ciphers. Since 1990 there has been a huge increase of public knowledge regarding the security of secret key block ciphers, most notably through the publication of the differential and linear attacks. Today we know how to break many systems faster than by an exhaustive search for the key. Still the best known attacks on many systems are not very practical and require either the encryptions of unrealisticly many chosen or known plaintexts and/or a huge memory and processing time.

References

1. K. Aoki and K. Ohta. Differential-linear attack on FEAL. *IEICE Trans. Fundamentals*, E79-A(1):20–27, 1996.

2. I. Ben-Aroya and E. Biham. Differential cryptanalysis of Lucifer. In D.R. Stinson, editor, *Advances in Cryptology: CRYPTO'93, LNCS 773*, pages 187–199, 1993.
3. T. Beth and C. Ding. On almost perfect nonlinear permutations. In T. Helleseth, editor, *Advances in Cryptology: EUROCRYPT'93, LNCS 765*, pages 65–76. Springer Verlag, 1993.
4. E. Biham. New types of cryptanalytic attacks using related keys. In T. Helleseth, editor, *Advances in Cryptology: EUROCRYPT'93, LNCS 765*, pages 398–409. Springer Verlag, 1993.
5. E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard*. Springer Verlag, 1993.
6. E. Biham and A. Shamir. Differential cryptanalysis of the full 16-round DES. In E.F. Brickell, editor, *Advances in Cryptology: CRYPTO'92, LNCS 740*, pages 487–496. Springer Verlag, 1993.
7. M. Blaze, W. Diffie, R.L. Rivest, B. Schneier, T. Shimomura, E. Thompson, and M. Wiener. Minimal key lengths for symmetric ciphers to provide adequate commercial security. Document, January 1996.
8. J.B. Borst, L.R. Knudsen, and V. Rijmen. Two attacks on IDEA. In *Advances in Cryptology: EUROCRYPT'97, LNCS*. Springer Verlag, 1997. To appear.
9. L. Brown, J. Pieprzyk, and J. Seberry. LOKI - a cryptographic primitive for authentication and secrecy applications. In J. Seberry and J. Pieprzyk, editors, *Advances in Cryptology: AusCrypt'90, LNCS 453*, pages 229–236. Springer Verlag, 1990.
10. P.M. Cohn. *Algebra, Volume 1*. John Wiley & Sons, 1982.
11. D. Coppersmith. The real reason for Rivest's phenomenon. In H.C. Williams, editor, *Advances in Cryptology: CRYPTO'85, LNCS 218*, pages 535–536. Springer Verlag, 1986.
12. T. Cusick and M. Wood. The REDOC-II cryptosystem. In A.J. Menezes and S.A. Vanstone, editors, *Advances in Cryptology: CRYPTO'90, LNCS 537*, pages 545–563. Springer Verlag, 1991.
13. J. Daemen, R. Govaerts, and J. Vandewalle. Weak keys for IDEA. In D.R. Stinson, editor, *Advances in Cryptology: CRYPTO'93, LNCS 773*, pages 224–231. Springer Verlag, 1993.
14. J. Daemen, L. Knudsen, and V. Rijmen. The block cipher SQUARE. In E. Biham, editor, *Fast Software Encryption, Fourth International Workshop, Haifa, Israel, January 1997, LNCS*. Springer Verlag, 1997. To appear.
15. I.B. Damgård and L.R. Knudsen. The breaking of the AR hash function. In T. Helleseth, editor, *Advances in Cryptology: EUROCRYPT'93, LNCS 773*, pages 286–292. Springer Verlag, 1993.
16. I.B. Damgård and L.R. Knudsen. Multiple encryption with minimum key. In E. Dawson and J. Golic, editors, *Cryptography: Policy and Algorithms. International Conference, Brisbane, Queensland, Australia, July 1995, LNCS 1029*, pages 156–164. Springer Verlag, 1995.
17. I.B. Damgård and L.R. Knudsen. Two-key triple encryption. *The Journal of Cryptology*, 1997. To appear.
18. D.W. Davies and W.L. Price. *Security for Computer Networks*. John Wiley & Sons, 1989.
19. D.E. Denning. *Cryptography and Data Security*. Addison-Wesley, 1982.
20. J. Detombe and S. Tavares. Constructing large cryptographically strong S-boxes. In J. Seberry and Y. Zheng, editors, *Advances in Cryptology: AusCrypt 92, LNCS 718*, pages 165–181. Springer Verlag, 1993.

21. W. Diffie and M. Hellman. New directions in cryptography. *IEEE Trans. on Information Theory*, IT-22(6):644–654, 1976.
22. S. Even and O. Goldreich. On the power of cascade ciphers. *ACM Trans. on Computer Systems*, 3:108–116, 1985.
23. H. Feistel. Cryptography and computer privacy. *Scientific American*, 228(5):15–23, 1973.
24. S. Goldwasser, S. Micali, and R.L. Rivest. A "paradoxical" solution to the signature problem. In *Proc. 25th IEEE Symposium on Foundations of Computer Science*, pages 441–448, 1984.
25. S. Goldwasser, S. Micali, and R.L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.
26. M. Hellman. A cryptanalytic time-memeory trade-off. *IEEE Trans. on Information Theory*, IT-26(4):401–406, 1980.
27. M.E. Hellman and S.K. Langford. Differential–linear cryptanalysis. In Y. Desmedt, editor, *Advances in Cryptology: CRYPTO'94, LNCS 839*, pages 26–39. Springer Verlag, 1994.
28. T. Jakobsen and L. Knudsen. The interpolation attack on block ciphers. In E. Biham, editor, *Fast Software Encryption, Fourth International Workshop, Haifa, Israel, January 1997, LNCS*. Springer Verlag, 1997. To appear.
29. D. Kahn. *The Codebreakers*. MacMillan, 1967.
30. B.S. Kaliski and M.J.B. Robshaw. Linear cryptanalysis using multiple approximations. In Y. Desmedt, editor, *Advances in Cryptology: CRYPTO'94, LNCS 839*, pages 26–39. Springer Verlag, 1994.
31. J. Kelsey, B. Schneier, and D. Wagner. Key-schedule cryptanalysis of IDEA, G-DES, GOST, SAFER, and triple-DES. In Neal Koblitz, editor, *Advances in Cryptology: CRYPTO'96, LNCS 1109*, pages 237–251. Springer Verlag, 1996.
32. J. Kilian and P. Rogaway. How to protect DES against exhaustive key search. In Neal Koblitz, editor, *Advances in Cryptology: CRYPTO'96, LNCS 1109*, pages 252–267. Springer Verlag, 1996.
33. L.R. Knudsen. Cryptanalysis of LOKI'91. In J. Seberry and Y. Zheng, editors, *Advances in Cryptology, AusCrypt 92, LNCS 718*, pages 196–208. Springer Verlag, 1993.
34. L.R. Knudsen. Cryptanalysis of LOKI. In H. Imai, R.L. Rivest, and T. Matsumoto, editors, *Advances in Cryptology: AsiaCrypt'91, LNCS 453*, pages 22–35. Springer Verlag, 1993.
35. L.R. Knudsen. *Block Ciphers – Analysis, Design and Applications*. PhD thesis, Aarhus University, Denmark, 1994.
36. L.R. Knudsen. Practically secure Feistel ciphers. In R. Anderson, editor, *Fast Software Encryption - Proc. Cambridge Security Workshop, Cambridge, U.K., LNCS 809*, pages 211–221. Springer Verlag, 1994.
37. L.R. Knudsen. A key-schedule weakness in SAFER K-64. In *Advances in Cryptology: CRYPTO'95, LNCS 963*, pages 274–286. Springer Verlag, 1995.
38. L.R. Knudsen. Truncated and higher order differentials. In B. Preneel, editor, *Fast Software Encryption - Second International Workshop, Leuven, Belgium, LNCS 1008*, pages 196–211. Springer Verlag, 1995.
39. L.R. Knudsen and T. Berson. Truncated differentials of SAFER. In Gollmann D., editor, *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 1996, LNCS 1039*, pages 15–26. Springer Verlag, 1995.

40. L.R. Knudsen and W. Meier. Improved differential attack on RC5. In Neal Koblitz, editor, *Advances in Cryptology: CRYPTO'96, LNCS 1109*, pages 216–228. Springer Verlag, 1996.
41. L.R. Knudsen and M.P.J. Robshaw. Non-linear approximations in linear cryptanalysis. In U. Maurer, editor, *Advances in Cryptology: EUROCRYPT'96, LNCS 1070*, pages 224–236. Springer Verlag, 1996.
42. X. Lai. On the design and security of block ciphers. In J.L. Massey, editor, *ETH Series in Information Processing*, volume 1. Hartung-Gorre Verlag, Konstanz, 1992.
43. X. Lai. Higher order derivatives and differential cryptanalysis. In R. Blahut, editor, *Communication and Cryptography, Two Sides of one tapestry*. Kluwer Academic Publishers, 1994. ISBN 0-7923-9469-0.
44. X. Lai, J.L. Massey, and S. Murphy. Markov ciphers and differential cryptanalysis. In D.W. Davies, editor, *Advances in Cryptology: EUROCRYPT'91, LNCS 547*, pages 17–38. Springer Verlag, 1992.
45. S.S. Magliveras and N.D. Memon. Algebraic properties of cryptosystem PGM. *Journal of Cryptology*, 5(3):167–184, 1992.
46. J.L. Massey. Cryptography: Fundamentals and applications. Copies of transparencies, Advanced Technology Seminars, 1993.
47. J.L. Massey. SAFER K-64: A byte-oriented block-ciphering algorithm. In R. Anderson, editor, *Fast Software Encryption - Proc. Cambridge Security Workshop, Cambridge, U.K., LNCS 809*, pages 1–17. Springer Verlag, 1994.
48. J.L. Massey. SAFER K-64: One year later. In B. Preneel, editor, *Fast Software Encryption - Second International Workshop, Leuven, Belgium, LNCS 1008*, pages 212–241. Springer Verlag, 1995.
49. M. Matsui. Linear cryptanalysis method for DES cipher. In T. Helleseth, editor, *Advances in Cryptology: EUROCRYPT'93, LNCS 765*, pages 386–397. Springer Verlag, 1993.
50. M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In Y.G. Desmedt, editor, *Advances in Cryptology: CRYPTO'94, LNCS 839*, pages 1–11. Springer Verlag, 1994.
51. M. Matsui. New structure of block ciphers with provable security against differential and linear cryptanalysis. In D. Gollman, editor, *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 1996, LNCS 1039*, pages 205–218. Springer Verlag, 1996.
52. M. Matsui and A. Yamagishi. A new method for known plaintext attack of FEAL cipher. In R. Rueppel, editor, *Advances in Cryptology: EUROCRYPT'92, LNCS 658*, pages 81–91. Springer Verlag, 1992.
53. U. Maurer and J.L. Massey. Cascade ciphers: The importance of being first. *Journal of Cryptology*, 6(1):55–61, 1993.
54. R. Merkle. Fast software encryption functions. In A.J. Menezes and S.A. Vanstone, editors, *Advances in Cryptology: CRYPTO'90, LNCS 537*, pages 476–501. Springer Verlag, 1991.
55. R. Merkle and M. Hellman. On the security of multiple encryption. *Communications of the ACM*, 24(7):465–467, 1981.
56. S. Miyaguchi. The FEAL cipher family. In A.J. Menezes and S.A. Vanstone, editors, *Advances in Cryptology: CRYPTO'90, LNCS 537*, pages 627–638. Springer Verlag, 1990.
57. S. Murphy, K. Paterson, and P. Wild. A weak cipher that generates the symmetric group. *Journal of Cryptology*, 7(1):61–65, 1994.

58. K. Nyberg. Differentially uniform mappings for cryptography. In T. Helleseth, editor, *Advances in Cryptology: EUROCRYPT'93, LNCS 765*, pages 55–64. Springer Verlag, 1993.
59. K. Nyberg. On the construction of highly nonlinear permutations. In R. Rueppel, editor, *Advances in Cryptology: EUROCRYPT'92, LNCS 658.* Springer Verlag, 1993.
60. K. Nyberg. Linear approximations of block ciphers. In A. De Santis, editor, *Advances in Cryptology: EUROCRYPT'94, LNCS 950*, pages 439–444. Springer Verlag, 1994.
61. K. Nyberg and L.R. Knudsen. Provable security against differential cryptanalysis. In E.F. Brickell, editor, *Advances in Cryptology: CRYPTO'92, LNCS 740*, pages 566–574. Springer Verlag, 1993.
62. K. Nyberg and L.R. Knudsen. Provable security against a differential attack. *The Journal of Cryptology*, 8(1):27–38, 1995.
63. National Bureau of Standards. Data encryption standard. Federal Information Processing Standard (FIPS), Publication 46, National Bureau of Standards, U.S. Department of Commerce, Washington D.C., January 1977.
64. National Bureau of Standards. DES modes of operation. Federal Information Processing Standard (FIPS), Publication 81, National Bureau of Standards, U.S. Department of Commerce, Washington D.C., December 1980.
65. B. Preneel. *Analysis and Design of Cryptographic Hash Functions.* PhD thesis, Katholieke Universiteit Leuven, January 1993.
66. B. Preneel, V. Rijmen, and A. Bosselaers. Recent developments in the design of conventional cryptographic algorithms, *This Volume*, pages 106–131.
67. V. Rijmen, J. Daemen, B. Preneel, A. Bosselaers, and E. De Win. The cipher SHARK. In Gollmann D., editor, *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 1996, LNCS 1039*, pages 99–112. Springer Verlag, 1996.
68. V. Rijmen, B. Preneel, E. De Win, On weaknesses of non-surjective round functions. *Designs, Codes, and Cryptography.* To appear.
69. R. Rivest. The RC5 encryption algorithm. In B. Preneel, editor, *Fast Software Encryption - Second International Workshop, Leuven, Belgium, LNCS 1008*, pages 86–96. Springer Verlag, 1995.
70. I. Schaumüller-Bichl. *Zur Analyse des Data Encryption Standard und Synthese Verwandter Chiffriersysteme.* PhD thesis, Linz University, May 1981.
71. I. Schaumüller-Bichl. On the design and analysis of new cipher systems related to the DES. Technical report, Linz University, 1983.
72. B. Schneier. Description of a new variable-length key, 64-bit block cipher (Blowfish). In R. Anderson, editor, *Fast Software Encryption - Proc. Cambridge Security Workshop, Cambridge, U.K., LNCS 809*, pages 191–204. Springer Verlag, 1994.
73. C.E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28:656–715, 1949.
74. A. Shimizu and S. Miyaguchi. Fast data encipherment algorithm FEAL. In D. Chaum and W.L. Price, editors, *Advances in Cryptology: EUROCRYPT'87, LNCS 304*, pages 267–280. Springer Verlag, 1988.
75. M.E. Smid and D.K. Branstad. The Data Encryption Standard: Past and future. In G.J. Simmons, editor, *Contemporary Cryptology - The Science of Information Integrity*, chapter 1, pages 43–64. IEEE Press, 1992.
76. A. Sorkin. LUCIFER: a cryptographic algorithm. *Cryptologia*, 8(1):22–35, 1984.
77. W. Tuchman. Hellman presents no shortcut solutions to DES. *IEEE Spectrum*, 16(7):40–41, July 1979.

78. P.C. van Oorschot and M.J. Wiener. A known-plaintext attack on two-key triple encryption. In Ivan B. Damgård, editor, *Advances in Cryptology: EUROCRYPT'90, LNCS 473*, pages 318–325. Springer Verlag, 1990.
79. P.C. van Oorschot and M.J. Wiener. Improving implementable meet-in-the-middle attacks of orders of magnitude. In Neal Koblitz, editor, *Advances in Cryptology: CRYPTO'96, LNCS 1109*, pages 229–236. Springer Verlag, 1996.
80. S. Vaudenay. On the weak keys of Blowfish. In Gollmann D., editor, *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 1996, LNCS 1039*, pages 27–32. Springer Verlag, 1996.
81. M.J. Wiener. Efficient DES key search. Technical Report TR-244, School of Computer Science, Carleton University, Ottawa, Canada, May 1994. Presented at the Rump Session of CRYPTO'93.