MODES OF BLOCKCIPHER ALGORITHMS AND THEIR PROTECTION AGAINST ACTIVE EAVESDROPPING

Cees J.A. Jansen[1] & Dick E. Boekee[2]

[1]Philips USFA B.V.
PO Box 218
5600 MD Eindhoven, The Netherlands

[2]Delft University of Technology
PO Box 5031
2600 GA Delft, The Netherlands

INTRODUCTION

Blockcipher algorithms are used in a variety of modes for message encryption or message authentication. The different modes not all offer the same protection against active eavesdropping. In this paper an overview of a number of modes and their protection against active eavesdropping is presented.

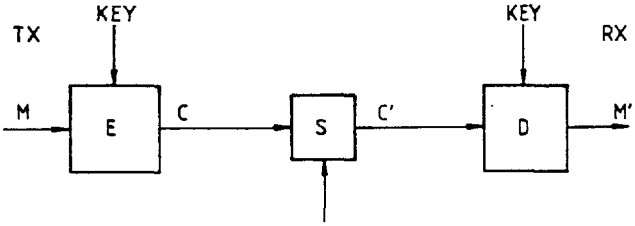
In figure 1 the problem of active eavesdropping (AE) is depicted. It is the objective of an active eavesdropper to manipulate the ciphertext C in such a way that a known message M is turned into a desired message M' without knowledge of the actual key that is being used. The manipulation operation S may consist of e.g. deleting, repeating or inserting parts of a message, but also of performing some arithmetic operation like addition of data.

The fact that M is known may be regarded as somewhat unrealistic, however it might be an authorized (standard) message or it might be a dummy message as is the case with traffic flow security (OSI).
So the receiver wonders wether M' is authentic and moreover may be confronted with random active eavesdropping in the form of bit-errors.

There are several methods known which offer protection against active eavesdropping, see e.g. [1]. One could use a message authentication code (MAC), but this is very sensitive to transmission errors, gives a certain text expansion, and gives a delayed notice. With delayed notice is meant that one has to wait a certain number of text blocks before one can possibly detect AE.

D. Chaum and W.L. Price (Eds.): Advances in Cryptology - EUROCRYPT '87, LNCS 304, pp. 281-286, 1988.

Figure 1:



One could also use some form of text feedback, but besides a certain delayed notice this always goes together with a phenomenon called error extension, which means that more than one text block will be erroneous if only one bit-error occurs. Finally one could use techniques for bitstreamciphers [2], but this can result in error extension beyond the block boundary even if only one bit-error occurs.

From the above it will be clear that we desire a method for protection against AE, applicable with blockcipher algorithms, which has no text expansion, no (block) error extension and the possibility of immediate notice. In section 1 various known and new modes and their behaviour under addition, deletion, repetition and insertion will be discussed. In section 2 implementations of the best mode with respect to protection against AE will be shown and their performance discussed.

SECTION 1

Let a blockcipher be given by its encryption and decryption operators E_k and D_k, which act on m-bit blocks under a key k. Blockcipher algorithms can be used in a variety of modes. Well known are the ECB, CBC, CFB and OFB modes [1], but there are more alternatives, such as PBC and PFB where we have interchanged the roles of message M and ciphertext C. Two new modes we have investigated are cipherblock chaining of message difference, CBCPD, and output feedback with a non-linear function, OFBNLF. These modes are depicted in figures 2 and 3 respectively.
The OFBNLF mode may be regarded as a combination of an OFB and an ECB mode and can in fact be implemented as such. However in section 2 it will be shown that an implementation can be much simpler.

Figure 2:

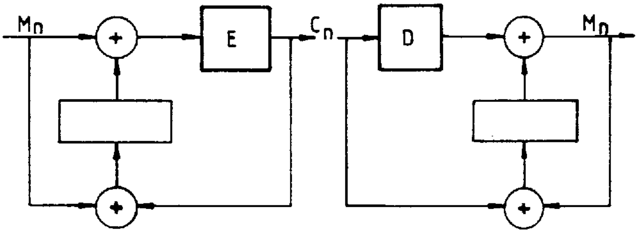
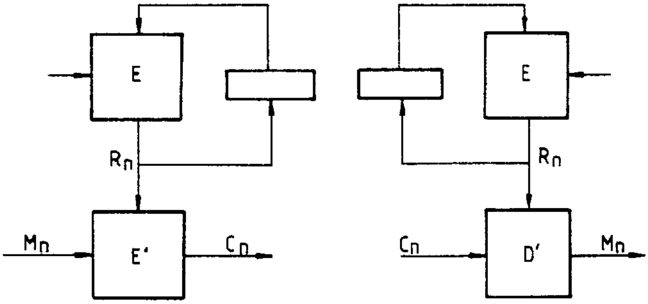


Figure 3:



The modes mentioned above are represented by the following equations:

ECB:	$C_n = E_k(M_n)$	$M_n = D_k(C_n)$
CBC:	$C_n = E_k(M_n + C_{n-1})$	$M_n = D_k(C_n) + C_{n-1}$
CFB:	$C_n = M_n + E_k(C_{n-1})$	$M_n = C_n + E\ (C_{n-1})$
OFB:	$C_n = M_n + R_n$ $R_n = E_k(R_{n-1})$	$M_n = C_n + R_n$
PBC:	$C_n = E_k(M_n) + M_{n-1}$	$M_n = D_k(C_n + M_{n-1})$
PFB:	$C_n = M_n + E_k(M_{n-1})$	$M_n = C_n + E_k(M_{n-1})$
CBCPD:	$C_n = E_k(M_n + M_{n-1} + C_{n-1})$	$M_n = D_k(C_n) + C_{n-1} + M_{n-1}$
OFBNLF:	$C_n = f_{R_n}(M_n)$ $R_n = E_k(R_{n-1})$	$M_n = f^{-1}_{R_n}(C_n)$

Here C_n and M_n denote the n^{th} ciphertext and plaintext blocks; R_n is the n^{th} block of pseudo-random bits and $f_{R_n}(M_n)$ is the R_n^{th} invertible function acting on plaintext block M_n.

From the equations one can easily see what happens if the n^{th} ciphertext block is deleted, repeated or added to some block S_n. As an example the deciphered message blocks in the case of deletion and addition are given below.

CIPHERTEXT :	C_{n-1}	C_{n+1}	C_{n+2}	.....
ECB :	M_{n-1}	M_{n+1}	M_{n+2}	.....
CBC / CFB :	M_{n-1}	?	M_{n+2}	.....
OFB / PBC / PFB / CBCPD / OFBNLF :	M_{n-1}	?	?	? ...

CIPHERTEXT :	C_{n-1}	C_n+ S	C_{n+1}	C_{n+2}	.....
OFB :	M_{n-1}	M_n+ S	M_{n+1}	M_{n+2}	.....
CFB :	M_{n-1}	M_n+ S	?	M_{n+2}	.....
PFB :	M_{n-1}	M_n+ S	?	?	? ...
CBC :	M_{n-1}	?	M_{n+1}+ S	M_{n+2}	.....
PBC / CBCPD :	M_{n-1}	?	?	?	? ...
ECB / OFBNLF:	M_{n-1}	?	M_{n+1}	M_{n+2}	.....

Here a ? denotes an unknown outcome of the decryption operation.

It can be seen that active eavesdropping will not be successfull with the OFBNLF mode, but occasional errors in the ciphertext will not give rise to block-error extension in the decrypted message.

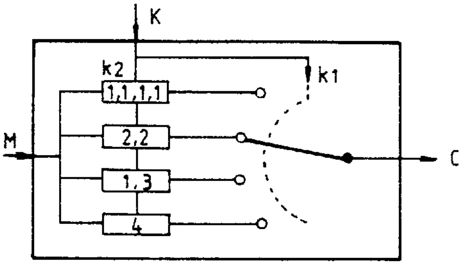
SECTION 2

As was already mentioned in section 1, the OFBNLF mode may be regarded as a combination of an OFB and an ECB mode. However the ECB part can be implemented in a much easier way. The purpose of the latter part is to keep the probability of success sufficiently low if an active eavesdropper performs some fixed transformation on a ciphertext block, such as some kind of addition. A solution which achieves this goal is to pseudo-randomly select a mixing function that mixes the message M and the blocks R. This function should be selected from a set of functions which are not all transparent for the same operation [2].

One example of this method is depicted in figure 4 where the the set of functions consists of four different real additions on 4-bit blocks, i.e. 4 bits modulo 2, 2 x 2 bits modulo 4, 1 bit modulo 2 and 3 bits modulo 8, and 4 bits modulo 16.

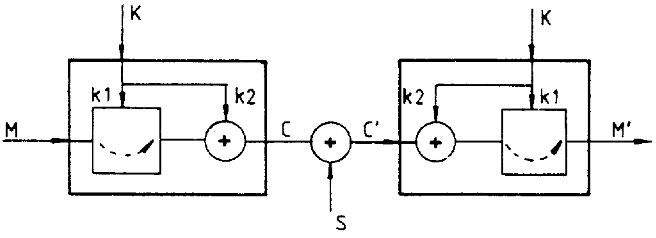
The performance in this case is rather difficult to evaluate, but it appears that the probability of success for an active eavesdropper, who is using real additions, tends to 50% for large blocklengths.

Figure 4:



A second example is depicted in figure 5, where the set of mixing functions consists of functions on the message blocks only, followed by a modulo 2 addition. As a specific set of message block functions cyclic shifts of the message blocks are chosen. Assuming that the active eavesdropper uses modulo 2 addition it can easily be seen that he has to guess the number cyclic shifts of the message block difference correctly.
I.e. $S = CYC_{K1}(M+M')$, where $CYC_k(X)$ denotes the block X shifted cyclically k times. If (M+M') has period P, then the probability of success is 1/P. By assuming equally likely message blocks the average successrate can be calculated. For example if the blocklength is 8 bits, the average successrate is :
$(1/1 + 2/2 + 12/4 + 240/8) / 255 = 11/85$ or 12.9%. Here we have obviously excluded the zero difference case, as this will always be successfull.

Figure 5:



The following table compares the average successrates of the cyclic shift and the real addition methods for some blocklengths.

BLOCKLENGTH	AVERAGE SUCCESSRATE CYCLIC SHIFTS	 REAL ADDITIONS
2	66.7 %	83.3 %
3	42.9 %	78.6 %
4	53.3 %	69.0 %
5	22.6 %	64.5 %
6	20.6 %	59.0 %
7	15.0 %	56.5 %
8	12.9 %	51.0 %

CONCLUSIONS

In this paper we have presented an overview of various modes of blockcipher algorithms and discussed their behaviour with respect to active eavesdropping. We also introduced a new mode called the OFBNLF mode, which offers good protection against active eavesdropping, has no text expansion, no block error extension, the possibility of immediate notice and can be implemented by means of simple operations.

REFERENCES

[1] C.H.Meyer & S.M.Matyas,"Cryptography",John Wiley & Sons, New York, 1982.

[2] C.J.A. Jansen,"Protection against active eavesdropping", EUROCRYPT 86, Linköping, Sweden, 1986.