VINO : A BLOCK CIPHER INCLÚDING VARIABLE PERMUTATIONS

Adina Di Porto, William Wolfowicz
Fondazione Ugo Bordoni, Rome, Italy

1. Introduction

In this paper a new secret key block cipher is proposed. Taking into account that in some works (e.g., see [1]) effective attacks to the DES have been devised, and considering the fact that block cipher algorithms are needed for national proprietary requirements, we developed the algorithm presented. In the proposed algorithm the input and the output are 64-bit blocks and the key is 128-bit long. Basically it belongs to the family of the modular arithmetic based algorithms, but furthermore a variable permutation is included. This is a very crucial point because permutations usually are very heavy to be implemented.

Our scheme makes use of a simple permutation property [2]. If this fact can reduce the number of possible permutations, on the other hand it makes both direct and inverse permutations easy to be implemented using simple software. Moreover the special kind of permutation which is included in our scheme depends both on the key and on the message. A mixture of the above operations and arithmetic ones seems to lead to an effective and strong algorithm.

In the sequel we will describe the elementary operations used and the building blocks of the algorithm. Moreover it must be taken into account that inversion is possible using the property shown in [2], and that the key-schedule is performed making use of the same permutations.

Even if some of the theorems included in [3] can be invoked to show that diffusion and confusion are achieved by the proposed scheme, a computer simulation has shown that these properties hold.

The name VINO, given to the algorithm comes from Vinogradov, whose theorem has been used to perform the keyed permutations. Thanks to Jim Massey for suggesting this name at the workshop.

2. Designing the algorithm

When we started designing the proposed algorithm, we had in mind that most of the algorithms can be classified as follows:

i) algorithms obtained modifying other existing algorithms;

ii) algorithms without any theoretical background;

iii) algorithms based on some theory.

Examples of algorithms belonging to the above listed categories can be found in the literature. Even if all the procedures can lead to strong and effective results, while the design as stated in ii) requires a special feeling about the discovery of flaws and weaknesses, the use of procedure iii) is always desirable, when possible. Nonetheless, working as in i) can be useful, by exploiting already well known properties of other existing algorithms, and trying to change those operations and procedures which give problems under every point of view. We leave the reader the decision about which class (classes) our algorithm belongs to.

Moreover, one of the main goals we wished to meet was a very high degree of flexibility in the use of the bits both of the message and of the key.

3. Defining V-Permutations

The implementation of keyed permutations is usually an operation heavy to be implemented. It requires the use of matrices which depend on the key and must be filled every time the permutation itself must be changed. Therefore, we were looking for a simpler way to permute binary sequences; an answer came unexpectedly from a theorem given as an exercise in the Vinogradov's book [2].

Theorem [2, p.121]:

" Let n be an integer, $n > 1$, and let m be an integer, $m > 1$. We consider the numbers 1, 2, ..., n in direct order from 1 to n, then in reverse order from n to 2, then in direct order from 1 to n, then in reverse order from n to 2, etc. From this sequence we take the 1-st, $(m + 1)$-st, $(2m + 1)$-st, etc., until we obtain n numbers. We repeat the same operation with this new sequence of n numbers, etc. Then the k-th operation gives the original sequence if and only if $m^k \equiv \pm 1 \pmod{2n - 1}$".

If we *decimate* (according to the procedure of the theorem) a n-bit string I_n, with step m $(1 \leq m \leq n - 1)$, thus obtaining I_n', then the above theorem enables us to obtain I_n by decimating I_n' with step $m^{k-1} \pmod{2n - 1}$.

In sec. 3.1 a small example will show how this theorem can be used to permute a sequence.

It must be pointed out that the use of this way of permuting lead to a small amount of permutations. Actually, if we have n bits, k ones and $(n - k)$ zeroes, the number of all the possible permutations is $\binom{n}{k}$, while Vinogradov's procedure gives only n permutations. To avoid this drawback we operate as follows (to simplify the exposition we take 16 as length of each sequence used):

let I be the sequence to be permuted, O the permuted sequence and H the key of the permutation; we subdivide H in four subblocks and we perform the following operations:

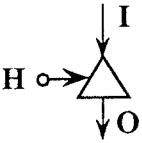
(1) *decimate* I with step m , as stated by the first four bits of H, giving I',

(2) cyclic shift on I' as stated by the second four bits of H, giving I'',

(3) operate (as in (1)) on I'' as stated by the third four bits of H, giving I''',

(4) operate (as in (2)) on I''' as stated by the last bits of H, giving the output O.

We will call it *V-Permutation* , and in the sequel we will denote it by



It must be noticed that all the above considerations hold with every lengths of I, O and H, provided that a suitable subblocking of H must be used.

It must be pointed out that we could not find a demonstration about the number of the possible V-Permutations, but computer simulations showed that this number is close, if not even equal, to the maximum.

3.1 *A small example*

By means of a small example we will try to show how the theorem of sec. 3. works.

Let 1 , 2 , 3 , 4 , 5 , 6 the sequence to be permuted, we have, in this case, $n = 6$ and let, for example, $m = 3$. Operating as stated by Vinogradov's theorem, we have

$$\underline{1}\ 2\ 3\ \underline{4}\ 5\ 6\ \underline{6}\ 5\ 4\ \underline{3}\ 2\ 1\ \underline{2}\ 3\ 4\ \underline{5}\ 6$$

thus obtaining 1 , 4 , 6 , 3 , 2 , 5 which is a permutation of the starting sequence.

To invert the permutation, it can be taken into account that

$3^{10} \equiv 1 \pmod{11}$, where $11 = 2n - 1$, and $3^5 \equiv 1 \pmod{11}$.

But having already decimated once, the inversion decimation step m' is given by

$$m' = 3^4 \equiv 4 \pmod{11}.$$

In fact, operating again according to Vinogradov, we have

$$\underline{1}\ 4\ 6\ 3\ \underline{2}\ 5\ 5\ 2\ \underline{3}\ 6\ 4\ 1\ \underline{4}\ 6\ 3\ 2\ \underline{5}\ 5\ 2\ 3\ \underline{6}\ 4\ 1$$

obtaining the starting sequence.

4. Generic round scheme

Let $\oplus$ denote bit-by-bit exclusive OR, and $\boxplus$ denote addition of integers modulo 2^n;

Considering a subblocking of the input and of the output of the generic r-th round into four 16-bit blocks and denoting by

- X_{ir} $(i = 1, 2, 3, 4)$ the input subblocks,
- Y_{ir} $(i = 1, 2, 3, 4)$ the output subblocks,
- K_{ir} $(i = 1, 2, 3,..., 8)$ the subblocks of the key used in the r-th round,

the generic round of the algorithm is represented in Fig. 1. Notice that in each round the input M_r is first V-permuted as stated by the first 64 bits of the round-key K_r.

5. Key schedule

The key length is 128 bits.

At the first round, the round-key is k_1, $k_1 = k$, where k is the secret initial key.

At each round the key k_j is scheduled according to the following procedure:

i) use k_{j-1} to V-permute k_{j-1} itself, subdividing it into 16 subblocks. Each of these subblocks indicate respectively decimation and cyclic shift in turn. Thus we get a variable that we call k'_{j-1},

ii) operating on k_{j-1} we perform a one bit left shift, leaving a zero on the last right position; then we complement every bit in position 0,2,4,... obtaining k''_{j-1},

iii) the desired key of the round is given by $k_j = k'_{j-1} \oplus k''_{j-1}$.

6. Final scheme

In the global algorithm scheme the message M is first encrypted using t rounds of the procedure shown in Fig.1, thus getting M'; as last step a final V-permutation is performed on M' giving the encrypted block C.

The choice of the number t of rounds is made mainly to avoid cryptanalitic attacks on the key. A suitable choice seems to be $t = 4$.

7.Conclusions

A new block cipher has been presented. The main advantages of this scheme are a very high degree of freedom in the use of the bits both of the key and of the message, the absence of weak keys and, last but not least, its cryptographic strength.

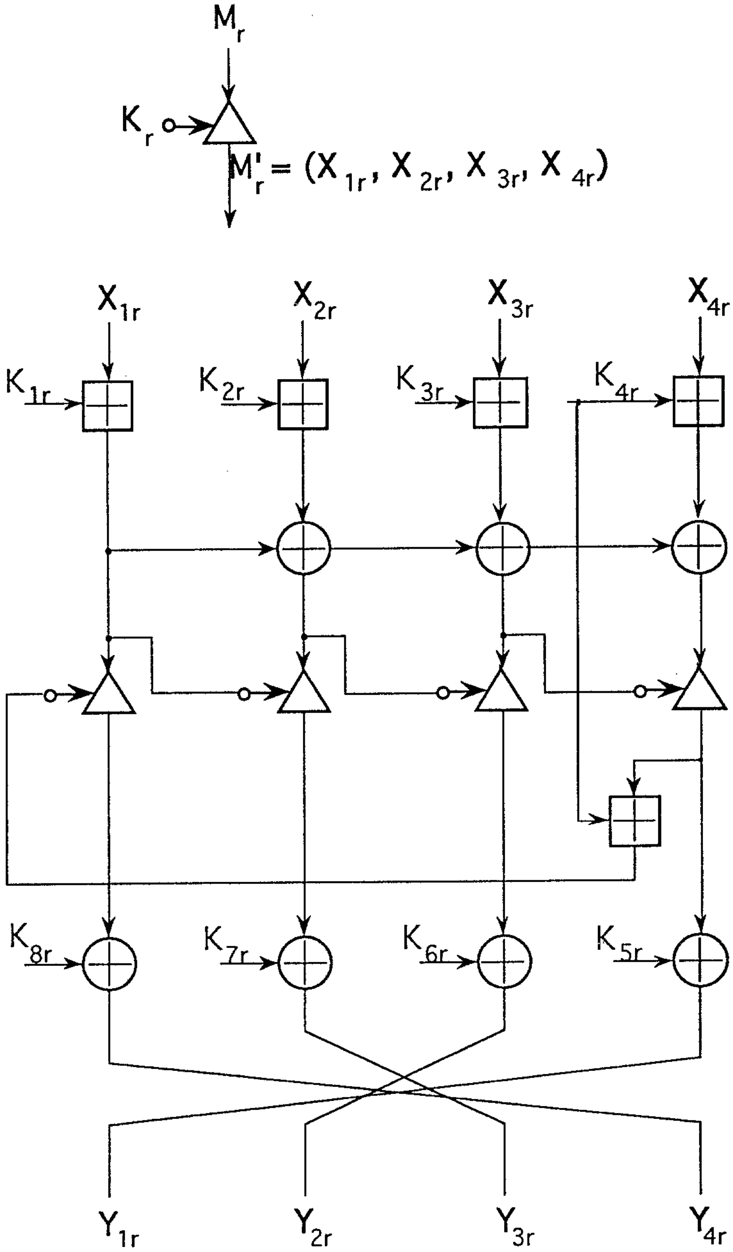

M_r
K_r
M'_r = (X_1r, X_2r, X_3r, X_4r)
X_1r
X_2r
X_3r
X_4r
K_1r
K_2r
K_3r
K_4r
K_8r
K_7r
K_6r
K_5r
Y_1r
Y_2r
Y_3r
Y_4r

Fig. 1

Acknowledgement

This work was carried out in the framework of an agreement between the Italian PT administration and the Fondazione Ugo Bordoni.

References

[1] M.J. Wiencr: " Efficient DES Key Search ", Crypto '93, Rump Session.

[2] I.M.Vinogradov: *Elements of number theory*, Dover Publications Inc., New York, 1945.

[3] X. Lai, J.L.Massey: "A Proposal for a New Block Encryption Standard" Advances in Cryptology-Eurocrypt '90, Springer-Verlag, Berlin 1991, pp.389-404.