# The Block Cipher Hierocrypt

Kenji Ohkuma[1], Hirofumi Muratani[1], Fumihiko Sano[2], and Shinichi Kawamura[1]

[1] Toshiba Corporate R & D Center
kenji.ohkuma@toshiba.co.jp
[2] Toshiba SI Thechnology Center

**Abstract.** This paper proposes a nested (hierarchical) SPN structure and the symmetric block cipher "Hierocrypt". In the nested SPN structure, lower-level SPN structures are recursively embedded into S-box positions in SPN of the higher level. This structure recursively assures the lower bound of active S-box number, and high security level is efficiently realized. The 8-round Hierocrypt is implemented in C language on Pentium III, and shows the middle-class performance of final AES candidates.

## 1 Introduction

The substitution-permutation network (SPN, for short) is one of the most important structures besides the Feistel network. The wide trail strategy is effective for an SPN cipher to achieve high security against the differential and linear cryptanalysis[1,2,3].

The optimal invertible linear mapping of diffusion layer is an essential component for the wide trail strategy. The mapping is usually called the maximum distance separable (MDS) mapping [2,4,5]. MDS mapping is optimal for the number of active S-boxes, which ensures the upper-bound of the characteristic probability for differential and linear cryptanalysis.

Rijmen et al. designed the 64-bit block cipher SHARK, where eight-parallel 8-bit S-boxes is mixed by the permutation layer, and the number of active S-boxes in the two consecutive layers is at least 9 [2]. It seems that the structure of SHARK is effective for a larger block size. But, straightforward extension to a larger block size has a disadvantage that the calculational cost for MDS part is proportional to the square of block size[1].

As a solution to the problem, Daemen, Rijmen et al. proposed the 128-bit ciphers SQUARE and Rijndael, where sixteen 8-bit S-boxes are divided into four parts composed of four S-boxes, and a local MDS operation is applied to each of them [4,5]. Although the minimum number of active S-boxes in consecutive two rounds is only 5, any trail of four consecutive rounds has at least 25 active S-boxes.

[1] The MDS operation consists of matrix multiplication where the number of matrix elements is proportional to the square of matrix size

D.R. Stinson and S. Tavares (Eds.): SAC 2000, LNCS 2012, pp. 72–88, 2001.

We propose a new class of SPN structure, a nested (or hierarchical) SPN structure, and the cipher Hierocrypt[2] based on the structure in this paper. The nested SPN structure is a multiple-level recursive structure (See Figure 1), and recursively assures the lower bound of active S-box number, and high security level is efficiently realized. The nested SPN structure can be regarded as an generalization of the SQUARE/Rijndael-type cipher. The generalization makes it possible to improve the security against the SQUARE attack.

The construction of this paper is as follows. In the following section, the nested SPN structure is introduced. In Sect. 3, we show an overview of the cipher Hierocrypt, which is composed of the nested SPN structure. In Sect. 4, we describe how the components of Hierocrypt are designed. In Sect. 5, the security of Hierocrypt against some attacks is discussed. Sect. 6 shows the software performance of 8-round Hierocrypt on some CPU. The final section is devoted to the concluding remarks.

## 2 Nested SPN Structure

A nested (hierarchical) SPN structure is a multiple-level structure, where an S-box in a certain level consists of a 1-level lower SPN (See Figure 1).

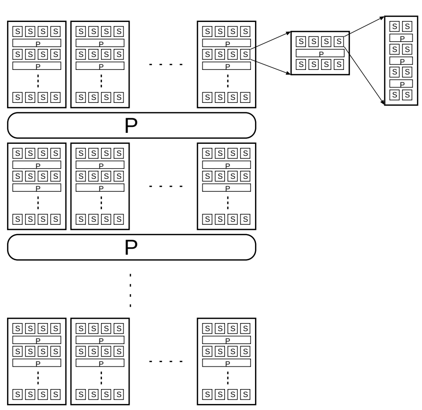


**Fig. 1.** Nested SPN structure

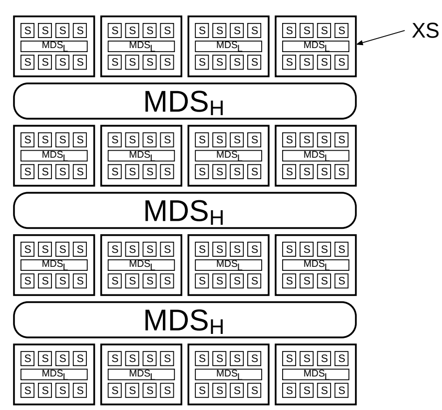


**Fig. 2.** 4-round netsted SPN ciphe

We impose the following conditions to realize the wide trail strategy efficiently [2,4,5].

**(a)** The final round of SPN consists only of an S-box layer (not followed by a diffusion layer) in all levels;
**(b)** All permutations are MDS in each level;
**(c)** The number of rounds is even in all levels except for the highest;

[2] Two versions of the cipher (Type-I and Type-II) is proposed in this paper

**(d)** Bit-wise key additions are located directly before all lowest-level S-box layers and directly after the final;

Figure 2 shows an example of two-level nested SPN cipher, which consists of 3 rounds of the higher-level SPN, whose S-boxes have a 2-round SPN structure. The number of parallel S-boxes is 4 for both levels. When the size of S-box is 8-bit, the block length of the cipher is 128-bit.

The following proposition is important to realize the wide trail strategy in the nested SPN.

**Proposition 1.** Consider 2-level nested SPN with the following conditions: (i) the above conditions (a)∼(c) are satisfied; (ii) the lower-level SPN is 2-round; (iii) the number of parallel S-boxes are $m_1$ and $m_2$ for the higher- and lower-level SPN, respectively. Then, any 2 consecutive higher-level rounds contain no less than $(m_1+1)(m_2+1)$ lower-level active S-boxes, for nonzero differential/mask.

*Proof.* For non-zero differential or non-zero mask pattern, there are no less than $(m_1+1)$ active higher-level S-boxes, each of which has no less than $(m_2+1)$ active lower-level S-boxes. Thus, at least $(m_1+1)(m_2+1)$ lower-level S-boxes are active. □

To construct a nested SPN cipher which satisfies the above condition (b), an MDS code with a large word-size is needed. Here, let the $(n,k,d)$ code be a code where $n$ is the block length, $k$ is the number of information digits, and $d$ is the minimum distance. For the case of Figure 2, we need (8,4,5) error-correcting code over 32-bit word set for the higher-level diffusion. Although the Reed-Solomon code over GF($2^{32}$) satisfies the condition, calculation over GF($2^{32}$) is often costly. An alternative way is construction by concatenating parallel smaller MDS-codes, which is based on the following proposition.

**Proposition 2.** Let $MDS_n$ be an MDS mapping defined by a $(2m,m,m+1)$-code over the $n$-bit word set. Then an MDS mapping $MDS_{m'n}$ based on $(2m,m,m+1)$-code over $m'n$-bit word set is constructed by concatenating $m'$ sets of $MDS_n$.

*Proof.* The proposition is proven by constructing an example of $(2m,m,m+1)$-code over $m'n$-bit words.

Consider $m'$ sets of mapping $MDS_n$.

$$MDS_n \ : \ x_{1j}\|x_{2j}\|\cdots\|x_{mj} \longmapsto y_{1j}\|y_{2j}\|\cdots\|y_{mj} \quad , \qquad 1 \le j \le m' \ ,$$

and define the concatenation as follows.

$$X_i = x_{i1}\|x_{i2}\|\cdots\|x_{im'} \quad , \quad Y_i = y_{i1}\|y_{i2}\|\cdots\|y_{im'} \quad , \qquad 1 \le i \le m \ .$$

Then define the following mapping $MDS_{m'n}$

$$MDS_{m'n} \ : \ X_1\|X_2\|\cdots\|X_m \longmapsto Y_1\|Y_2\|\cdots\|Y_m \quad .$$

For nonzero differential/mask, at least one of the $m'$ sub-mappings $MDS_n$ is active, therefore at least $(m+1)$ $m'n$-bit words from $\{X_i\}$ and $\{Y_i\}$ are active. This means that the mapping is MDS. □

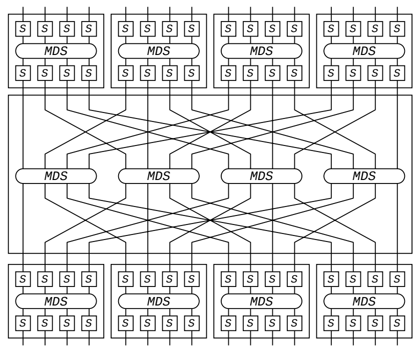


**Fig. 3.** SQUARE in nested SPN form

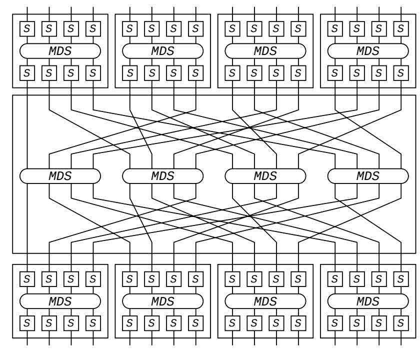


**Fig. 4.** Rijndael in nested SPN form

The above construction of $MDS_{m'n}$ is a fundamental one. The mapping is generalized by putting invertible linear transformations on input and output words of $m'n$-bit length.

The above construction of MDS mapping gives a new viewpoint for SQUARE and Rijndael. Figures 3 and 4 respectively show mathematically equivalent forms of four-round SQUARE and Rijndael. All MDS are the same. The central large rectangle corresponds to the MDS of higher level, consisting of 4 parallel sub-MDS of $(8, 4, 5)$-code. Thus, the parameters are $m = m' = 4$ ; $n = 8$, which guarantees that no less than 25 S-boxes are active in four consecutive rounds.

That is, the same code is used in both-level MDS mapping for SQUARE.

## 3 Overview of Hierocrypt Encryption

Hierocrypt is a two-level nested SPN cipher with which has the following features.

($\alpha$) The size of lower-level S-box is 8-bit;
($\beta$) The number of parallel S-boxes is 4 in both levels;
($\gamma$) Lower-level structure is 2-round SPN;
($\delta$) Diffusion layers of both levels consist of MDS mapping defined by (8, 4, 5)-code.

The lower-level diffusion $MDS_{\mathrm{L}}$ is based on (8, 4, 5)-code over GF($2^8$). Two types of higher-level diffusions $MDS_{\mathrm{H}}$ are designed on (8, 4, 5)-codes over GF($2^{32}$) and over GF($2^4$), respectively.

For respective Galois fields, the following primitive polynomials are used in this paper.

$$
\begin{aligned}
p_4(z) &= z^4 + z + 1 \ , && \text{for GF}(2^4) \ , \\
p_8(z) &= z^8 + z^6 + z^5 + z + 1 \ , && \text{for GF}(2^8) \ , \\
p_{32}(z) &= z^{32} + z^{28} + z^{27} + z + 1 \ , && \text{for GF}(2^{32}) \ .
\end{aligned}
$$

### 3.1 Lower-Level Components of Hierocrypt

The lower-level SPN structure of Hierocrypt consists of three components: bit-wise key addition $AK$, byte substitution $[S]$, and diffusion $[MDS_{\mathrm{L}}]$.

**Bit-Wise Key Addition $\boldsymbol{AK}$ (1-Bit×128).** 128-bit half-round key $K$ is added to the data (See Sect. 3.4 for the key scheduling).

$$AK(K) \ : \ X \longmapsto Y \iff Y = X \oplus K \ .$$

**Byte Substitution $[\boldsymbol{S}]$ (8-Bit×16).** The byte substitution $[S]$ consists of parallel operations of S-box $S$ (See Sect. 4.1).

$$[S] \ : \ X \longmapsto Y \iff y_{ij} = S(x_{ij}) \ , \quad i = 1, \ldots, 4 \ ; \ j = 1, \ldots, 4 \ .$$

The square bracket [ ] means parallel operation.

**Lower-Level Diffusion $[\boldsymbol{MDS}_{\mathrm{L}}]$ (32-Bit×4).** The lower-level diffusion $MDS_{\mathrm{L}}$ mixes 4 parallel bytes (See Sect. 4.2).

$$[MDS_{\mathrm{L}}] \ : \ X \longmapsto Y \iff Y_i = MDS_{\mathrm{L}}(X_i) \ , \quad i = 1, \ldots, 4 \ .$$

### 3.2 Higher-Level Components of Hierocrypt

The higher-level SPN consists of two components: parallel 4-byte substitution $[XS]$ and higher-level diffusion $MDS_{\mathrm{H}}$, except for the final key addition.

**4-Byte Substitution $[\boldsymbol{XS}]$ (32-Bit×4).**

$$[XS] \ : \ X \longmapsto Y \iff Y_i = XS(X_i) \ , \qquad i = 1, \ldots, 4 \ .$$

4-byte substitution $[XS]$ consists of the three kinds of lower-level components, combined as follows.

$$[XS] = [S] \circ AK\left(K^{\beta}\right) \circ [MDS_{\mathrm{L}}] \circ [S] \circ AK\left(K^{\alpha}\right) \ ,$$

where $K^{\alpha}$ and $K^{\beta}$ is the first and the second half of round key (See Sect. 3.4).

**Higher-Level Diffusion $\boldsymbol{MDS}_{\mathrm{H}}$ (128-Bit×1).**

$$MDS_{\mathrm{H}} \ : \ X \longmapsto Y \ .$$

Two types are given in Sect. 4.3 and 4.3.

### 3.3 Round Functions and Encryption

**Round Function $\boldsymbol{\rho}$ (Except for Final).**

$$\rho = MDS_{\mathrm{H}} \circ [XS] \ .$$

**Final Round Function $\rho'$.**

$$\rho' = [XS] \ .$$

**Hierocrypt Encryption.** The Hierocrypt encryption of $T$ rounds consists of $(T-1)$ iterations of round function $\rho$ followed by the final round function $\rho'$ and the final key addition.

$$Enc = AK(K^{(T+1)\alpha}) \ \circ \ \rho'(K^{(T)}) \ \circ \ \rho(K^{(T-1)}) \ \circ \ \cdots \ \circ \ \rho(K^{(2)}) \ \circ \ \rho(K^{(1)}) \ .$$

### 3.4 Key Scheduling

The key scheduling part consists of an initial key expansion $KX$ and iterative key generations $KH$.

$$\begin{aligned} K^{(0)} &= KX\,(K) \ , \\ K^{(t)} &= KH\ \left(K^{(t-1)}\right) \ , \qquad (1 \leq t \leq 2T+1) \ . \end{aligned}$$

The data randomization part requires two-round iterations of $KH$ per round.

**Initial Key Expansion.** The initial key expansion $KX$ expands an encryption key $K$ (128/192/256 bits) up to 256-bit by padding. The 32-bit key data $K_i$ is represented as concatenation of four 8-bit data.

$$K_i = k_{i1} \| k_{i2} \| k_{i3} \| k_{i4} \ .$$

**[128-bit key]**

$$\begin{aligned} K &= K_1 \| K_2 \| K_3 \| K_4 \\ &= k_{11} \| k_{12} \| k_{13} \| k_{14} \| k_{21} \| k_{22} \| \cdots \| k_{43} \| k_{44} \ . \\ K^{(0)} &= K_1 \| K_2 \| K_3 \| K_4 \| K_1 \| K_2 \| K_3 \| K_4 \ . \end{aligned}$$

**[192-bit key]**

$$\begin{aligned} K &= K_1 \| K_2 \| K_3 \| K_4 \| K_5 \| K_6 \\ &= k_{11} \| k_{12} \| \cdots \| k_{43} \| k_{44} \| k_{51} \| k_{52} \| \cdots \| k_{63} \| k_{64} \ . \\ K^{(0)} &= K_1 \| K_2 \| K_3 \| K_4 \| K_5 \| K_6 \| K_1 \| K_2 \ . \end{aligned}$$

**[256-bit key]**

$$K^{(0)} = K \ = \ K_1 \| K_2 \| K_3 \| K_4 \| K_5 \| K_6 \| K_7 \| K_8 \ .$$

**Key Round Function.** The key round function transforms the $(t-1)$-th intermediate key $K^{(t-1)}$ into the $t$-th one $K^{(t)}$,

$$K^{(t)} = KH \left(K^{(t-1)}\right) \ .$$

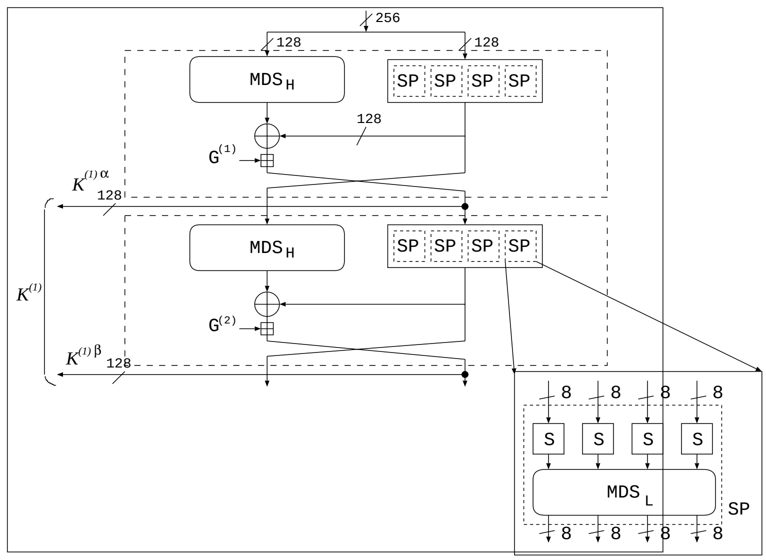


**Fig. 5.** Overview of the key scheduling

The key $K^{(t)}$ divides into two halves: $KL^{(t)}$ (left half) and $KR^{(t)}$ (right half),

$$K^{(t)} = KL^{(t)} \| KR^{(t)} \ .$$

The output of key round function corresponds to the round key for data randomization as follows.

$$K^{(t)\alpha} = KR^{(2t-1)} \ , \qquad K^{(t)\beta} = KR^{(2t)} \ .$$

The outline of the key round function is as follows (See Fig.5).

• **[right $\longmapsto$ left]**

$$[MDS_{\mathrm{L}}] \circ [S] \ : \qquad KR^{(t-1)} \ \longmapsto \ KL^{(t)} \ .$$

where $[S]$ is given in Sect. 3.1.2, and $[MDS_{\mathrm{L}}]$ is given in Sect. 3.1.3.

• **[left $\longmapsto$ right]**

$$AD_{+}\left(G^{(t)}\right) \circ AD\left(KL^{(t)}\right) \circ MDS_{\mathrm{H}} \ : \qquad KL^{(t-1)} \ \longmapsto \ KR^{(t)} \ .$$

where $MDS_{\mathrm{H}}$ is given in Sect. 3.2.2, $AD_{+}\left(G^{(t)}\right)$ is the addition of round constant $G^{(t)}$.

$$KR_i^{(t)} = KR_i'^{(t)} + G_i^{(t)} \qquad (\bmod\ 2^{32}) \ , \qquad (i = 1, \ldots, 4) \ .$$

The constants $G^{(t)}$ are given in Table 1.

**Table 1.** Round constant $G^{(t)}$

| | |
|---|---|
| $G^{(1)}$ | $= (H_2, H_0, H_1, H_1)$ |
| $G^{(2)}$ | $= (H_3, H_2, H_0, H_3)$ |
| $G^{(3)}$ | $= (H_1, H_0, H_0, H_0)$ |
| $G^{(4)}$ | $= (H_1, H_0, H_1, H_3)$ |
| $G^{(5)}$ | $= (H_0, H_1, H_0, H_2)$ |
| $G^{(6)}$ | $= (H_3, H_2, H_0, H_0)$ |
| $G^{(7)}$ | $= (H_1, H_2, H_1, H_0)$ |
| $G^{(8)}$ | $= (H_2, H_1, H_2, H_3)$ |
| $G^{(9)}$ | $= (H_2, H_1, H_0, H_0)$ |
| $G^{(10)}$ | $= (H_1, H_1, H_1, H_2)$ |
| $G^{(11)}$ | $= (H_3, H_1, H_1, H_2)$ |
| $G^{(12)}$ | $= (H_1, H_1, H_2, H_0)$ |
| $G^{(13)}$ | $= (H_1, H_3, H_3, H_1)$ |
| $G^{(14)}$ | $= (H_2, H_3, H_3, H_1)$ |
| $G^{(15)}$ | $= (H_1, H_3, H_1, H_0)$ |
| $G^{(16)}$ | $= (H_1, H_0, H_0, H_3)$ |
| $G^{(17)}$ | $= (H_1, H_2, H_0, H_3)$ |

**Generation of Round Constant $G^{(t)}$.** To prevent the weak key generation such as a cyclic pattern in round key sequence, we introduce 32-bit constant parameters $G_i^{(t)}(t = 1, \ldots, 17;\ i = 1, \ldots, 4)$, which are given by Table 1. Here, the constants $H_i(i = 1, \ldots, 4)$ is given as follows (The prefix "0x" indicates hexadecimal numbers).

$$H_0 = \text{0x5A827999} = trunc\left(\sqrt{2}/4\right), \qquad H_1 = \text{0x6ED9EBA1} = trunc\left(\sqrt{3}/4\right),$$

$$H_2 = \text{0x8F1BBCDC} = trunc\left(\sqrt{5}/4\right), \quad H_3 = \text{0xCA62C1D6} = trunc\left(\sqrt{10}/4\right),$$

where '*trunc*' is the truncation function which is defined by using the floor function[3].

$$trunc(x) = \lfloor 2^{32} x \rfloor \ .$$

Table 1 is given by the following simple rule. We use the eight-bit linear feedback shift register (LFSR) of Fibonacci type, of which the primitive polynomial is $z^8 + z^4 + z^3 + z^2 + 1$, and the initial state is $z^7 + z^3 + z + 1$. The LFSR generates a bit sequence $\zeta_1, \zeta_2, \zeta_3, \ldots$. Successive two bits of them determines the suffix of $H_j$. Specifically, the $i$-th constant of the $t$-th round $G_i^{(t)}$ is given as follows.

$$G_i^{(t)} = H_{\phi(t,i)}\ , \qquad \phi(t,i) = 2\zeta_{8(t-1)+2i-1} + \zeta_{8(t-1)+2i}\ .$$

## 4 Design of the Components

We describe how the components of Hierocrypt are designed in this section.

[3] The floor function $\lfloor x \rfloor$ is the largest integer no more than $x$

In the component design, the maximum differential and linear probabilities are the most important security measures for block ciphers. The maximum differential probability for the function $f$ is defined as

$$dp^f \equiv \max_{\Delta x \neq 0, \Delta y} \frac{\#\left\{x | f(x) \oplus f(x \oplus \Delta x) = \Delta y\right\}}{2^n} . \qquad (1)$$

Similarly, the maximum linear probability for the function $f$ is given as

$$lp^f \equiv \max_{\Gamma x, \Gamma y \neq 0} \left| \frac{\#\left\{x | x \cdot \Gamma x = f(x) \cdot \Gamma y\right\}}{2^n} - \frac{1}{2} \right| . \qquad (2)$$

### 4.1 Lower-Level S-Box $S$

The lower-level S-box $S$ is given as follows (in hexadecimal expression).

$$
\begin{array}{l}
(S(0)\ S(1)\ \cdots\ S(\mathrm{F})\ S(10)\ \cdots\ S(\mathrm{FF})) \\
= (\ \mathtt{72\ AA\ 49\ 16\ 1E\ 3A\ 43\ AE\ 66\ BC\ 00\ 73\ 79\ 3B\ FB\ 9F} \\
\quad \mathtt{69\ 6A\ A2\ 50\ 6E\ F5\ EF\ AC\ 22\ 02\ AD\ 26\ E2\ DF\ 97\ F0} \\
\quad \mathtt{9E\ BF\ 17\ 8B\ FA\ 7C\ F4\ 71\ 7F\ CA\ F6\ 52\ FD\ C3\ E5\ 64} \\
\quad \mathtt{53\ 8D\ E0\ F3\ 0F\ 78\ CB\ 9B\ 68\ 3C\ 0D\ 1F\ 89\ B6\ EB\ F7} \\
\quad \mathtt{44\ 4A\ 06\ A6\ 56\ 6B\ 85\ 01\ 30\ 88\ 51\ 31\ 9C\ A0\ A3\ 25} \\
\quad \mathtt{60\ 5B\ FF\ 05\ B7\ 91\ 15\ B3\ A9\ 20\ 03\ 2B\ 61\ 42\ 95\ 4D} \\
\quad \mathtt{F9\ 7E\ 0E\ E9\ D8\ F1\ 46\ 99\ CE\ BE\ D9\ 54\ 80\ B0\ D2\ 4F} \\
\quad \mathtt{7A\ E8\ 35\ 92\ 1B\ 7B\ 12\ D6\ 4C\ D5\ E7\ EE\ B1\ 24\ DE\ 21} \\
\quad \mathtt{04\ 10\ AB\ 29\ 9A\ 81\ FE\ A7\ B8\ 63\ 28\ 0A\ 8A\ D1\ C6\ 07} \\
\quad \mathtt{B9\ C8\ 98\ 82\ 74\ 9D\ 84\ 47\ 94\ C7\ 6C\ 11\ D7\ BA\ C1\ C9} \\
\quad \mathtt{DD\ 77\ 39\ 2F\ 2E\ C2\ 67\ 41\ E4\ 58\ 34\ CD\ 1C\ 93\ 96\ 7D} \\
\quad \mathtt{2C\ F8\ B5\ 70\ 14\ 08\ DC\ CC\ 87\ D0\ 5E\ 32\ C5\ C4\ 59\ 3E} \\
\quad \mathtt{CF\ 55\ 5C\ 23\ 75\ 2D\ 2A\ 86\ 4B\ 1D\ 5F\ E6\ FC\ B2\ 4E\ 09} \\
\quad \mathtt{27\ AF\ 19\ B4\ BD\ 6D\ 3D\ 6F\ ED\ 62\ EA\ F2\ D3\ 36\ 38\ DB} \\
\quad \mathtt{BB\ 83\ 45\ 37\ A4\ EC\ 8C\ 5D\ E1\ 33\ 90\ A1\ 40\ 8E\ 1A\ A5} \\
\quad \mathtt{0B\ 3F\ 5A\ DA\ 13\ 76\ 0C\ C0\ 48\ E3\ 65\ A8\ 18\ 8F\ D4\ 57}\ ) .
\end{array}
$$

The maximum differential probability of the lower-level S-box $S$ is

$$dp^S = \frac{6}{256} .$$

And the distribution of differential probabilities for nonzero input differentials is shown in Table 2.

Similarly, for the maximum linear probability,

$$lp^S = \frac{22}{256} .$$

And the distribution of the linear probabilities for nonzero output mask patterns is shown in Table 3.

The algebraic order of the S-box is 7-th, which is the highest value for 8-bit bijection.

**Table 2.** Differential probability distribution

| differential probability (/256) | num. |
|---|---|
| 6 | 50 |
| 4 | 1,899 |
| 2 | 28,692 |
| 0 | 34,639 |
| total | 65,280 |

**Table 3.** Linear probability distribution

| linear probability (/256) | num. |
|---|---|
| 22 | 20 |
| 20 | 157 |
| 18 | 577 |
| 36 | 579 |
| ⋮ | ⋮ |
| 0 | 5,748 |
| total | 65,280 |

### 4.2 Lower-Level Diffusion $MDS_{\mathrm{L}}$ and Higher-Level S-Box $XS$

The security of higher-level S-box $XS$ depends on the combination of S-box $S$ and permutation $MDS_{\mathrm{L}}$. We have chosen $MDS_{\mathrm{L}}$ from randomly generated MDS matrices over GF($2^8$), such that the differential and linear properties are better than the usual case where all active S-boxes take the worst probability.

As the branch number of $MDS_{\mathrm{L}}$ is 5, the maximum differential and linear characteristic probabilities: $DP^{XS}$ and $LP^{XS}$, respectively satisfy the following inequalities.

$$DP^{XS} \leq \left(\frac{6}{256}\right)^5 \cong 2^{-27.1} \ ,$$

$$LP^{XS} \leq \frac{1}{2}\left(2 \cdot \frac{22}{256}\right)^5 \cong 2^{-13.7} \ .$$

The above inequalities are satisfied, only if the lower-level permutation $MDS_{\mathrm{L}}$ is an MDS mapping. We succeeded in improving the differential and linear properties by selecting the following $MDS_{\mathrm{L}}$.

$$MDS_{\mathrm{L}}(X_i) = D_{\mathrm{L}} X_i \ ,$$

$$D_{\mathrm{L}} = \begin{pmatrix} \mathrm{6C} & 25 & \mathrm{9B} & 03 \\ \mathrm{6D} & 06 & \mathrm{C8} & 18 \\ 75 & 78 & \mathrm{9E} & \mathrm{1F} \\ 42 & 78 & \mathrm{EB} & 61 \end{pmatrix} \ .$$

The matrix elements are expressed in hexadecimal and regarded as elements of GF($2^8$). For example, the elements 25 is regarded as the polynomial $z^5+z^2+1$ of GF($2^8$).

$$0x25 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$$
$$\Longleftrightarrow 1 \cdot z^5 + 0 \cdot z^4 + 0 \cdot z^3 + 1 \cdot z^2 + 0 \cdot z^1 + 1 \cdot z^0 = z^5 + z^2 + 1 \ .$$

We use the following criteria for selection.

- At least one S-box has a differential probability no more than (4/256), for any nonzero input differential of which active S-box number is 5 (See Table 2).
- At least one S-box has a linear probability no more than (20/256), for any nonzero output mask pattern of which active S-box number is 5 (See Table 3).

If $MDS_{\mathrm{L}}$ satisfies the criteria, the inequalities for $XS$ are refined as follows.

$$DP^{XS} \le \frac{4}{256}\left(\frac{6}{256}\right)^4 \cong 2^{-27.7} \ ,$$
$$LP^{XS} \le \frac{1}{2}\left(2 \cdot \frac{20}{256}\right)\left(2 \cdot \frac{22}{256}\right)^4 \cong 2^{-13.8} \ .$$

### 4.3 Higher-Level Diffusion $MDS_{\mathrm{H}}$

The higher-level diffusion $MDS_{\mathrm{H}}$ is based on $(8, 4, 5)$-code over 32-bit words. We give two examples, [Type-I]: mapping based on $(8, 4, 5)$-code over GF($2^{32}$); [Type-II]: concatenation of 8 parallel mappings based on $(8, 4, 5)$-code over GF($2^4$).

**Type-I $MDS_{\mathrm{H}}$.** Type-I $MDS_{\mathrm{H}}$ is selected from randomly generated matrices over GF($2^{32}$). For calculational efficiency, we impose the condition that only lowest 5 bits can be nonzero for all matrix elements. Multiplication with a constant over GF($2^{32}$) reduces to 4 times of table-lookup where the respective inputs are 8-bit long.

$$MDS_{\mathrm{H}}(X) = D_{\mathrm{H}}X \ ,$$
$$D_{\mathrm{H}} = \begin{pmatrix} 05 & 19 & 06 & 1B \\ 1B & 05 & 19 & 06 \\ 06 & 1B & 05 & 19 \\ 19 & 06 & 1B & 05 \end{pmatrix} \ .$$

**Type-II $MDS_{\mathrm{H}}$.** Type-II $MDS_{\mathrm{H}}$ is based on the following $4 \times 4$ matrix $D_{\mathrm{h}}$, which is selected from randomly generated matrices over GF($2^4$).

$$MDS_{\mathrm{H}}(X) = D_{\mathrm{H}}X \ , \quad D_{\mathrm{H}} = D_{\mathrm{h}} \otimes I_8 \ ,$$
$$D_{\mathrm{h}} = \begin{pmatrix} 6 & B & D & C \\ C & 6 & B & D \\ D & C & 6 & B \\ B & D & C & 6 \end{pmatrix}$$

Here, $I_8$ means 8-dimensional identity matrix. $D_\mathrm{h} \otimes I_8$ is eight parallel multiplications of the matrix $D_\mathrm{h}$ to the 16-bit vector (regarded as four elements of $\mathrm{GF}(2^4)$) which is made by picking up one bit from each byte.

The byte-oriented form of mapping is as follows.

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \end{pmatrix} = \begin{pmatrix} 0&1&1&0&0&1&0&1&0&0&1&1&1&0&1&1 \\ 1&0&1&1&1&0&1&0&0&0&0&1&0&1&0&1 \\ 0&1&0&1&1&1&0&1&1&0&0&0&1&0&1&0 \\ 1&1&0&0&1&0&1&1&0&1&1&1&0&1&1&0 \\ 1&0&1&1&0&1&1&0&0&1&0&1&0&0&1&1 \\ 0&1&0&1&1&0&1&1&1&0&1&0&0&0&0&1 \\ 1&0&1&0&0&1&0&1&1&1&0&1&1&0&0&0 \\ 0&1&1&0&1&1&0&0&1&0&1&1&0&1&1&1 \\ 0&0&1&1&1&0&1&1&0&1&1&0&0&1&0&1 \\ 0&0&0&1&0&1&0&1&1&0&1&1&1&0&1&0 \\ 1&0&0&0&1&0&1&0&0&1&0&1&1&1&0&1 \\ 0&1&1&1&0&1&1&0&1&1&0&0&1&0&1&1 \\ 0&1&0&1&0&0&1&1&1&0&1&1&0&1&1&0 \\ 1&0&1&0&0&0&0&1&0&1&0&1&1&0&1&1 \\ 1&1&0&1&1&0&0&0&1&0&1&0&0&1&0&1 \\ 1&0&1&1&0&1&1&1&0&1&1&0&1&1&0&0 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \\ x_{31} \\ x_{32} \\ x_{33} \\ x_{34} \\ x_{41} \\ x_{42} \\ x_{43} \\ x_{44} \end{pmatrix}$$

# 5 Security

## 5.1 Differential and Linear Cryptanalysis

The branch number is 5 for both mappings $MDS_\mathrm{H}$ and $MDS_\mathrm{L}$, Thus, Hierocrypt contains at least 5 active $XS$ per two rounds, which contain at least 5 active S-boxes, for a non-zero differential/mask. Therefore, Hierocrypt contains at least 25 active S-boxes per two rounds (See Proposition 1).

Then, the maximum differential and linear characteristics respectively satisfy the following inequalities.

$$DP^{2\,\mathrm{rounds}} = \left(\frac{6}{256}\right)^{25} \cong 2^{-135.3} \ll 2^{-128} \, .$$

$$LP^{2\,\mathrm{rounds}} \le \frac{1}{2}\left(2 \cdot \frac{22}{256}\right)^{25} \cong 2^{-64.5} < 2^{-64} \, .$$

This estimation is for the general $MDS_\mathrm{L}$. The characteristics for $MDS_\mathrm{L}$ in Sect. 4.2 is a little better (Sect. 4.2).

$$DP^{XS} \le 2^{-27.7} \, , \quad LP^{XS} \le 2^{-13.8} \, ,$$

$$DP^{2\,\mathrm{rounds}} \le \left(2^{-27.7}\right)^5 \cong 2^{-138.3} \ll 2^{-128} \, ,$$

$$LP^{2\,\mathrm{rounds}} \le \frac{1}{2}\left(2 \cdot 2^{-13.8}\right)^5 \cong 2^{-65.2} < 2^{-64} \, .$$

The above result shows that all two-round differential characteristics are far below the critical noise level $2^{-128}$, and that the maximum 2-round linear characteristic is equal order of the critical noise level $2^{-64}$. Then, we recommend to use Hierocrypt with no less than 6 rounds, where intermediate 4 rounds are for sufficiently small characteristics, and 2 remaining rounds on both ends against partial exhaustive key search.

### 5.2 Other Cryptanalysis

Besides the differential and the linear cryptanalysis discussed in the previous subsection, there are many other attacks. We discuss the security against the SQUARE (dedicated) attack and the higher-order differential cryptanalysis here [4,8].

The SQUARE attack is presented by J.Daemen, L.R.Knudsen, and V.Rijmen in their paper proposing the block cipher SQUARE [4]. The attack is applicable to SQUARE and other ciphers with similar SPN structures, such as Rijndael or CRYPTON [5,9]. The attack is applicable to Hierocrypt as well, because of its SPN structure. Here, we regard one round of Hierocrypt as two "half-rounds" to compare the strength with SQUARE[4].

The basic version of SQUARE attack is applicable to 4-round SQUARE and 4-round Rijndael. The attack is based on the following property, that all (16) bytes of the 3rd round output are always balanced over the $\Lambda$-set input with only one active byte. Thus, the 128 key bits can be identified by the basic attack with $2^9$ plaintexts and $2^9$ encryption time for SQUARE and Rijndael.

On the other hand, only two bytes of the 3rd half-round output are always balanced on the same condition. This property reduces to the fact that only 64 key bits can be identified by the basic attack with $2^{11}$ plaintexts and $2^{11}$ encryption time for Hierocrypt(Type-II). Thus, the efficiency of basic attack for Hierocrypt(Type-II) is 1/8 of that for SQUARE/Rijndael.

Further study shows that Hierocrypt is stronger in any extended versions of the dedicated attack.

Next, we discuss about the higher differential cryptanalysis. It is known that the cryptanalysis is applicable to $\mathcal{KN}$cipher, which is provably secure against the differential and linear cryptanalysis[8]. The security against the higher differential cryptanalysis is estimated by the algebraic order. The algebraic order of Hierocrypt's S-box is 7. And the order after 3 S-box layers (1.5 layers in Hierocrypt convention) is roughly estimated as

$$7^3 = 343 \gg 128 \ .$$

Therefore, the cryptanalysis does not seem to be feasible.

## 6 Performance

The cipher Hierocrypt is implemented in four Microprocessors. The measured encryption rates are shown in Table 4.

As the implementation here is rather basic one, the performance is expected to improve by optimizing the implementation.

[4] One half-round of Hierocrypt corresponds to one round of usual SPN cipher, as it contains one S-box layer

**Table 4.** Speed of Type-I Hierocrypt encryption(8 rounds)

| Processor | Freq | Platform | Compiler | Throughput |
|---|---|---|---|---|
| Pentium III | 550MHz | WindowsNT 4.0 | VC++ 6.0 | 43.18 Mbps |
| Celeron | 466MHz | Linux kernel-2.2.5 | egcs 1.1.2 | 31.74 Mbps |
| Ultra SPARC II | 296MHz | Solaris 2.5.1 | gcc 2.95.1 | 15.90 Mbps |
| Alpha 21164A | 599MHz | Digital UNIX V4.0D | gcc 2.95.1 | 27.04 Mbps |

**Table 5.** Speed of Type-II Hierocrypt encryption(8 rounds)

| Processor | Freq | Platform | Compiler | Throughput |
|---|---|---|---|---|
| Pentium III | 550MHz | WindowsNT 4.0 | VC++ 6.0 | 40.33 Mbps |
| Celeron | 466MHz | Linux kernel-2.2.5 | egcs 1.1.2 | 29.00 Mbps |
| Ultra SPARC II | 296MHz | Solaris 2.5.1 | gcc 2.95.1 | 19.10 Mbps |
| Alpha 21164A | 599MHz | Digital UNIX V4.0D | gcc 2.95.1 | 48.00 Mbps |

## 7 Concluding Remarks

We propose the block encryption algorithm "Hierocrypt" based on a two-level nested SPN structure. We use MDS mappings for both-level permutations, which assures the minimum number of active S-boxes hierarchically. SQUARE and Rijndael can be regarded as the nested SPN ciphers, where the higher- and lower-level diffusion layers ($MDS_{\mathrm{H}}$ and $MDS_{\mathrm{L}}$) are made by using the same MDS matrix. On the other hand, the diffusion layers of both levels are designed independently for Hierocrypt. This independency is profitable to improve the security against many attacks including the SQUARE dedicated attack.

## Appendix

## A Improved Algorithm

We have designed a revised version named "Hierocrypt-3" based on the Type-II algorithm. All components except for the nested SPN structure of the data randomization part are modified. The following modified components of data randomization are presented in this Appendix.

1. S-box
2. Higher-Level Diffusion $MDS_{\mathrm{H}}$
3. Lower-Level Diffusion $MDS_{\mathrm{L}}$

### A.1 S-Box

The new S-box $S(x)$ is given by the following table in hexadecimal notation.

$$(S(0)\ S(1)\ \cdots\ S(\mathrm{F})\ S(10)\ \cdots\ S(\mathrm{FF}))$$

$$
= \begin{pmatrix}
07 & FC & 55 & 70 & 98 & 8E & 84 & 4E & BC & 75 & CE & 18 & 02 & E9 & 5D & 80 \\
1C & 60 & 78 & 42 & 9D & 2E & F5 & E8 & C6 & 7A & 2F & A4 & B2 & 5F & 19 & 87 \\
0B & 9B & 9C & D3 & C3 & 77 & 3D & 6F & B9 & 2D & 4D & F7 & 8C & A7 & AC & 17 \\
3C & 5A & 41 & C9 & 29 & ED & DE & 27 & 69 & 30 & 72 & A8 & 95 & 3E & F9 & D8 \\
21 & 8B & 44 & D7 & 11 & 0D & 48 & FD & 6A & 01 & 57 & E5 & BD & 85 & EC & 1E \\
37 & 9F & B5 & 9A & 7C & 09 & F1 & B1 & 94 & 81 & 82 & 08 & FB & C0 & 51 & 0F \\
61 & 7F & 1A & 56 & 96 & 13 & C1 & 67 & 99 & 03 & 5E & B6 & CA & FA & 9E & DF \\
D6 & 83 & CC & A2 & 12 & 23 & B7 & 65 & D0 & 39 & 7D & 3B & D5 & B0 & AF & 1F \\
06 & C8 & 34 & C5 & 1B & 79 & 4B & 66 & BF & 88 & 4A & C4 & EF & 58 & 3F & 0A \\
2C & 73 & D1 & F8 & 6B & E6 & 20 & B8 & 22 & 43 & B3 & 33 & E7 & F0 & 71 & 7E \\
52 & 89 & 47 & 63 & 0E & 6D & E3 & BE & 59 & 64 & EE & F6 & 38 & 5C & F4 & 5B \\
49 & D4 & E0 & F3 & BB & 54 & 26 & 2B & 00 & 86 & 90 & FF & FE & A6 & 7B & 05 \\
AD & 68 & A1 & 10 & EB & C7 & E2 & F2 & 46 & 8A & 6C & 14 & 6E & CF & 35 & 45 \\
50 & D2 & 92 & 74 & 93 & E1 & DA & AE & A9 & 53 & E4 & 40 & CD & BA & 97 & A3 \\
91 & 31 & 25 & 76 & 36 & 32 & 28 & 3A & 24 & 4C & DB & D9 & 8D & DC & 62 & 2A \\
EA & 15 & DD & C2 & A5 & 0C & 04 & 1D & 8F & CB & B4 & 4F & 16 & AB & AA & A0
\end{pmatrix} .
$$

$$s(x) = Add(Power(Perm(x))) .$$

where *Perm* is a bit permutation

$$
\begin{aligned}
Perm \ &: \mathrm{GF}(2)^8 \longmapsto \mathrm{GF}(2)^8 , \\
y_i &= x_{\pi(i)} ,
\end{aligned}
$$

**Table 6.** Bit permutation for S-box

| i | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $\pi(i)$ | 3 | 7 | 5 | 8 | 6 | 2 | 4 | 1 |

*Power* is the power of 247 over $\mathrm{GF}(2^8)$ with the primitive polynomial $z^8 + z^6 + z^5 + z + 1$.

$$
\begin{aligned}
s \ &: \mathrm{GF}(2^8) \longmapsto \mathrm{GF}(2^8) , \\
s(x) &= x^{247} ,
\end{aligned}
$$

*Add* is a constant addition.

$$
\begin{aligned}
Perm \ &: \mathrm{GF}(2)^8 \longmapsto \mathrm{GF}(2)^8 , \\
Add(x) &= x \oplus 0\mathrm{x}11 .
\end{aligned}
$$

The power function *Power* has the same probabilities, both the bit permutation *Perm* and the constant addition *Add* do not change them.

*Perm* is chosen so that the number of polynomials of output bits is the maximum, in order to improve the security against the interpolation attack.

*Add* is chosen so that the distribution of input and output hamming distances is nearest to that of random function, in order to remove the statistical bias.

The main purpose of S-box modification is to improve the security against the differential and linear cryptanalysis. The maximum differential and linear probabilities of the S-box are $2^{-6}$ and $2^{-4}$, respectively, which are proven to be minimum theoretically. The maximum differential and linear characteristic probabilities for two rounds are respectively estimated as $2^{-150}$ and $2^{-76}$ when the above S-box is used. This estimation indicates that both differential and linear characteritic probabilities saturate after two rounds.

### A.2 Lower-Level Diffusion $MDS_\mathrm{L}$

The lower-level diffusion $MDS_\mathrm{L}$ is given as follows.

$$MDS_\mathrm{L} \ : \ \mathrm{GF}(2^8)^4 \longmapsto \mathrm{GF}(2^8)^4 \ ,$$
$$\begin{pmatrix} y_{i1} \\ y_{i2} \\ y_{i3} \\ y_{i4} \end{pmatrix} = \begin{pmatrix} \mathrm{C4} & 65 & \mathrm{C8} & \mathrm{8B} \\ \mathrm{8B} & \mathrm{C4} & 65 & \mathrm{C8} \\ \mathrm{C8} & \mathrm{8B} & \mathrm{C4} & 65 \\ 65 & \mathrm{C8} & \mathrm{8B} & \mathrm{C4} \end{pmatrix} \begin{pmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ x_{i4} \end{pmatrix}$$

We have chosen this maxtrix from circulant MDS matrices so that the output of SP-function (composite function of S-box and $MDS_\mathrm{L}$) has the maximum number of polynomials.

### A.3 Higher-Level Diffusion $MDS_\mathrm{H}$

The higher-level diffusion $MDS_\mathrm{H}$ is given as follows.

$$MDS_\mathrm{H}(X) = D_\mathrm{H} X \ , \quad D_\mathrm{H} = D_\mathrm{h} \otimes I_8 \ ,$$
$$D_\mathrm{h} = \begin{pmatrix} 5 & 5 & \mathrm{A} & \mathrm{E} \\ \mathrm{E} & 5 & 5 & \mathrm{A} \\ \mathrm{A} & \mathrm{E} & 5 & 5 \\ 5 & \mathrm{A} & \mathrm{E} & 5 \end{pmatrix} \ ,$$
$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \end{pmatrix} = \begin{pmatrix}
1&0&1&0&1&0&1&0&1&1&0&1&1&1&1&1 \\
1&1&0&1&1&1&0&1&1&1&1&0&0&1&1&1 \\
1&1&1&0&1&1&1&0&1&1&1&1&0&0&1&1 \\
0&1&0&1&0&1&0&1&1&0&1&0&1&1&1&0 \\
1&1&1&1&1&0&1&0&1&0&1&0&1&1&0&1 \\
0&1&1&1&1&1&0&1&1&1&0&1&1&1&1&0 \\
0&0&1&1&1&1&1&0&1&1&1&0&1&1&1&1 \\
1&1&1&0&0&1&0&1&0&1&0&1&1&0&1&0 \\
1&1&0&1&1&1&1&1&1&0&1&0&1&0&1&0 \\
1&1&1&0&0&1&1&1&1&1&0&1&1&1&0&1 \\
1&1&1&1&0&0&1&1&1&1&1&0&1&1&1&0 \\
1&0&1&0&1&1&1&0&0&1&0&1&0&1&0&1 \\
1&0&1&0&1&1&0&1&1&1&1&1&1&0&1&0 \\
1&1&0&1&1&1&1&0&0&1&1&1&1&1&0&1 \\
1&1&1&0&1&1&1&1&0&0&1&1&1&1&1&0 \\
0&1&0&1&1&0&1&0&1&1&1&0&0&1&0&1
\end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \\ x_{31} \\ x_{32} \\ x_{33} \\ x_{34} \\ x_{41} \\ x_{42} \\ x_{43} \\ x_{44} \end{pmatrix} .$$

The MDS matrix is chosen so that any byte is connected to all bytes after one round (one $MDS_\mathrm{L}$ and one $MDS_\mathrm{H}$) through more than one intermediate bytes. This condition is imposed to improve the security against the SQUARE dedicated attack. Our evaluation shows that the condition makes the nested SPN cipher at least one half-round stronger than SQUARE and Rijndael against the attack.

## References


1. S.Vaudenay, "On the need for multipermutations": Cryptanalysis of MD4 and SAFER. Fast Software Encryption(2), LNCS **1008**, pp.286-297, 1995.
2. V.Rijmen, J.Daemen, B.Preneel, A.Bosselaers, E.DcWin, "The Cipher SHARK," Fast Software Encryption(3), LNCS **1039**, pp.99-112, 1996.
3. A.M.Youssef, S.Mister, and S.E.Tavres, "On the Design of Linear Transformations for Substitution Permutation Encryption Networks," Selected Areas in Cryptography, SAC'97, Workshop Report, http://saturn.ee.queensu.ca:8000/sac/sac97/papers/paper29.ps.
4. J.Daemen, L.R.Knudsen, V.Rijmen, "The block cipher Square," Fast Software Encryption(4), LNCS **1267**, pp.149-165, 1997.
5. J.Daemen, V.Rijmen, "AES Proposal: Rijndael," http://www.esat.kuleuven.ac.be/∼rijmen/rijndael/rijndaeldocV2.zip
6. E.Biham and A.Shamir, "Differential cryptanalysis of DES-like cryptosystems," Journal of Cryptology, **4** No.1, pp.3-72, 1991.
7. M.Matsui, "Linear cryptanalysis method for DES cipher," Eurocrypt'93, LNCS **765**, pp.386-397, 1994.
8. L.R. Knudsen, "Truncated and higher order differentials," Fast Software Encryption(2), LNCS **1008**, pp.196-211, 1995.
9. C.D'Halluin, G. Bijnens, V. Rijmen, B. Preneel, "Attack on 6 rounds of Crypton," Fast Software Encryption(6), LNCS.1636, pp.46-59, 1999.