The Synchronous 8th-Order Differential Attack on 12 Rounds of the Block Cipher HyRAL

Yasutaka Igarashi, Seiji Fukushima, and Tomohiro Hachino
Kagoshima University, Kagoshima, Japan
Email: {igarashi, fukushima, hachino}@eee.kagoshima-u.ac.jp
Naoki Shibayama
Japan Air Self-Defense Force, Tokyo, Japan
shiba@wit.ocn.ne.jp
Toshinobu Kaneko
Tokyo University of Science, Tokyo, Japan
Email: kaneko@ee.noda.tus.ac.jp

Abstract—We study the synchronous 8th-order differential attack on the 128-bit block cipher HyRAL proposed by Hirata of Laurel Intelligent Systems in 2010. HyRAL supports 128, 192, and 256 bits of secret keys. We found the new synchronous 8th-order differential characteristics of HyRAL in the 8th round of data-mixing part. Exploiting the characteristics we show that 12 rounds of HyRAL can be attacked with 2^{128} blocks of chosen plain text and 2^{233.7} times of data encryption. We have reduced data complexity to 1/2^8 and reduced computational complexity to 1/2^{13} compared to the conventional attack.

Index Terms—cryptanalysis, higher-order differential attack, block cipher, HyRAL

I. INTRODUCTION

HyRAL is a 128-bit block cipher proposed by Hirata of Laurel Intelligent Systems in 2010 [1]. It supports 128, 192, and 256 bits of secret keys. HyRAL consists of byte-wise swaps, nonlinear layers, and linear layers. It has been reported that a data-mixing function of HyRAL is secure against differential attacks and linear attacks [2], [3]. It has been also reported that 12 rounds of HyRAL out of 32 rounds can be attacked with the 16th-order differential characteristics, 2^{209} blocks of chosen plain text and 2^{233.7} times of data encryption [4].

In this article, we show the new synchronous 8th-order differential characteristics of HyRAL in the 8th round, and show that 12 rounds of HyRAL can be attacked by exploiting the characteristics we found with low complexities. In Section V we conclude our article.

II. DATA-MIXING FUNCTION OF HYRAL

Fig. 1 shows data-mixing functions of HyRAL for a 128-bit key (a) and for a 192-bit or 256-bit key (b). They consist of 128-bit input/output (I/O) functions, G_1, G_2, F_1, and F_2. RK_i (i = 1, 2, ..., 9) and IK_j (j = 1, 2, ..., 6) are 128-bit sub keys. The symbol "⊕" indicates an XOR operation. X^{(0)} denotes a 128-bit plain text. X^{(20)} in (a) and X^{(52)} in (b) denote 128-bit cipher texts.

Fig. 2 and Fig. 3 show G_1, G_2, F_1, and F_2 where X_i = X_i^{(10)} || X_i^{(11)} || X_i^{(12)} || X_i^{(13)} and IK_i = IK_{i0} || IK_{i1} || IK_{i2} || IK_{i3}. X_j^{(j)} (j = 1, 2, 3, 4) and IK_j (j = 0, 1, 2, 3) are 32-bit data. The symbol "‖" denotes a concatenation of two data. G_i and F_j (j = 1, 2) consist of 4 iteration rounds, which consists of XOR and f_i with 32-bit I/O (i = 1, 2, ..., 8).

Fig. 4 shows f_i, which consists of the byte-wise swap, S, P, and XOR. The symbol “0x” represents its following value is hexadecimal. Assuming the I/O of the swap as 32-bit vectors x and x', where x = x_0 || x_1 || x_2 || x_3, x' is given by

\[
\begin{align*}
x'_0 &= x_0 \oplus x_1 \oplus x_2 \oplus x_3, \\
x'_1 &= x_2 \oplus x_1 \oplus x_0, \\
x'_2 &= x_2 \oplus x_1 \oplus x_0 \oplus x_3, \\
x'_3 &= x_2 \oplus x_1 \oplus x_0 \\
x'_4 &= x_2 \oplus x_1 \oplus x_0 \\
x'_5 &= x_2 \oplus x_1 \oplus x_0 \\
x'_6 &= x_2 \oplus x_1 \oplus x_0 \\
x'_7 &= x_0 \oplus x_3 \oplus x_2, \\
x'_8 &= x_2 \oplus x_1 \oplus x_0 \\
x'_9 &= x_2 \oplus x_1 \oplus x_0 \\
x'_10 &= x_2 \oplus x_1 \oplus x_0 \\
x'_11 &= x_2 \oplus x_1 \oplus x_0 \oplus x_3 \\
x'_12 &= x_2 \oplus x_1 \oplus x_0 \\
x'_13 &= x_2 \oplus x_1 \oplus x_0 \\
x'_14 &= x_2 \oplus x_1 \oplus x_0 \\
x'_15 &= x_2 \oplus x_1 \oplus x_0 \\
x'_16 &= x_2 \oplus x_1 \oplus x_0 \\
x'_17 &= x_2 \oplus x_1 \oplus x_0 \\
x'_18 &= x_2 \oplus x_1 \oplus x_0 \\
x'_19 &= x_2 \oplus x_1 \oplus x_0 \\
x'_20 &= x_2 \oplus x_1 \oplus x_0 \\
x'_21 &= x_2 \oplus x_1 \oplus x_0 \\
x'_22 &= x_2 \oplus x_1 \oplus x_0 \\
x'_23 &= x_2 \oplus x_1 \oplus x_0 \\
x'_24 &= x_2 \oplus x_1 \oplus x_0 \\
x'_25 &= x_2 \oplus x_1 \oplus x_0 \\
x'_26 &= x_2 \oplus x_1 \oplus x_0 \\
x'_27 &= x_2 \oplus x_1 \oplus x_0 \\
x'_28 &= x_2 \oplus x_1 \oplus x_0 \\
x'_29 &= x_2 \oplus x_1 \oplus x_0 \\
x'_30 &= x_2 \oplus x_1 \oplus x_0 \\
x'_31 &= x_2 \oplus x_1 \oplus x_0
\end{align*}
\]

x_j (j = 0, 1, 2, 3) is an 8-bit data. S denotes the S-box with 8-bit I/O, which is a bijective nonlinear function. P is a 4 × 4 non-singular matrix given by

Manuscript received April 15, 2013; revised July 17, 2013.
The multiplications of a matrix and a vector are performed in GF($2^8$) defined by the characteristic polynomial $z^8 + z^4 + z^3 + z + 1$. $f_i$ can be equivalently modified as shown in Fig. 5 where the number in a box represents multiplication and $y_{ij}$ ($i, j = 0, 1, 2, 3$) represents an 8-bit intermediate data of $f_i$.

III. HIGHER-ORDER DIFFERENTIAL

In this section, we describe the definition of higher-order differential and some of its properties related to this article [5], and we describe an attack equation using these properties.

Fig. 6 shows a block diagram of an encryption process. $E_1$ and $E_2$ represent components of an encryption process. $K_1 \in GF(2^p)$ and $K_2 \in GF(2^q)$ represent $p$ bits and $q$ bits of the extended keys used in $E_1$ and $E_2$, respectively. $P = (p_1, p_2, ..., p_n)$ and $\Delta P \in GF(2^n)$ represent $n$ bits of input plain text and input difference, respectively. $H \in GF(2^m)$ represents $m$ bits of the output of $E_1$. $C(P \oplus \Delta P)$
\[ \Delta^{(j)}E_{K}^{-1}(C(P); K_{2}) = 0 \] (7)

Equations (6) and (7) are always correct if \( K_{2} \) is correct, while they are stochastically correct if \( K_{2} \) is incorrect. This is why attacker can estimate \( K_{2} \) and check the correctness of \( K_{2} \) by (6) or (7). The incorrect \( K_{2} \) can be eliminated by solving some sets of the equation (6) or (7) whose plain texts \( P \) are different from each other. Actually, we have to solve at least \( \lceil q/m \rceil \) different sets of (6) or (7). Such attack using (6) or (7) is called a higher-order differential attack. Equation (6) or (7) is called an attack equation.

IV. THE SYNCHRONOUS 8TH-ORDER DIFFERENTIAL CHARACTERISTICS OF HYRAL AND ITS ATTACK EQUATION

In this section, we describe the new synchronous 8th-order differential characteristics of HyRAL we found, and describe the attack equation exploiting the characteristics. At the end we estimate the number of chosen plain texts and the number of encryption operations required to identify the sub keys. Fig. 7 shows the new synchronous 8th-order differential path of HyRAL where \( f_{1} \) in the 8th round is equivalently modified. Note that \( RK_{i} = RK_{00} \parallel RK_{01} \parallel RK_{02} \parallel RK_{03} \) where \( RK_{j} \) (\( j = 0, 1, 2, 3 \)) is a 32-bit sub key. \( P^{-1} \) is the inverse matrix of \( P \). \( C \) is the constant addition.

\[ X^{(0)} = (X_{1}^{(0)} X_{2}^{(0)} X_{3}^{(0)} X_{4}^{(0)}) \]

Figure 6. Block diagram of an encryption process.

Figure 7. The synchronous 8th-order differential path of HyRAL.

Since \( E_{ij}(P; K_{j}) = E_{2}^{-1}(C(P); K_{2}) \), which is the inverse function of \( E_{2} \), (4) and (5) can be rewritten as

\[ \Delta^{(N+1)}E_{i}^{-1}(C(P); K_{i}) = 0 \] (4)

Moreover, if the Boolean polynomial of \( E_{i}(P; K_{i}) \) does not include the \( j \)-th order term as \( \prod_{l \neq i} p_{l} \) (1 \( \leq t \leq n \)), the \( j \)-th order differential of \( E_{i}(P; K_{i}) \) becomes zero regardless of \( P \) and \( K_{i} \) as follow

\[ \Delta^{(j)}E_{i}(P; K_{i}) = 0 \] (5)

Finally, the output cipher text \( \{ \Delta P \} \) is determined by \( E_{2}^{-1}(C(P); K_{2}) \).
difference as the upper 8 bits of \( X_3^{(0)} \) is put into the upper 8 bits of \( X_4^{(0)} \) shown as “S” in Fig. 7, which we call “synchronous.” The remaining 14 bytes shown as “C” in Fig. 7 are set to take arbitrary constants. In this case, we found that the 8th-order differentials of 32-bit A and 32-bit B in the 8th round take the same value regardless of a plain text and the sub keys in the previous rounds. By exploiting the characteristics we can derive the following attack equation as

\[
\sum_{\Delta X^{(0)}} A_8 = \sum_{\Delta X^{(0)}} B_8 \tag{8}
\]

\[
A_8 = S(f_6((X_2^{(9)} \oplus R K'_{30})^{\delta_2} \oplus (X_2^{(10)})^{\delta_6} \oplus (IK'_{13})^{\delta_8})) \tag{9}
\]

\[
B_i = P^{-1}(X_2^{(11)} \oplus C)^{\delta_8} \tag{10}
\]

\[
X_2^{(9)} = f_6(f_6((X_2^{(11)} \oplus R K'_{32}) \oplus X_2^{(12)} \oplus I K'_{33} ) \oplus X_2^{(13)}) \tag{11}
\]

\[
X_2^{(10)} = f_6(f_6(X_2^{(12)} \oplus R K'_{33}) \oplus X_2^{(13)} \oplus I K'_{34}) \oplus X_2^{(14)} \tag{12}
\]

\[
X_2^{(11)} = f_6(f_6(X_2^{(12)} \oplus R K'_{34}) \oplus X_2^{(13)} \oplus I K'_{35}) \oplus X_2^{(15)} \tag{13}
\]

\[
X_2^{(12)} = (X_2^{(10)} \oplus \Delta X^{(0)}) = X_2^{(12)} \parallel X_2^{(12)} \parallel X_2^{(12)} \tag{14}
\]

\[
R K'_{30} = R K'_{30} \oplus R K'_{40}, \quad I K'_{13} = I K'_{13} \oplus R K'_{33} \oplus R K'_{43} \tag{15}
\]

\[
I K'_{21} = I K'_{21} \oplus R K'_{41}, \quad I K'_{22} = I K'_{22} \oplus R K'_{40} \tag{16}
\]

\[
I K'_{23} = I K'_{23} \oplus R K'_{41} \tag{17}
\]

where \( A_8 \) and \( B_8 \) denote the upper first bytes of A and B in Fig. 7, respectively. \( (\cdot)^{\delta} \) denotes the upper \( \delta \)th byte of data x. C denotes the constant addition in Fig. 7. \( X_2^{(12)}(X_2^{(9)} \oplus A X^{(0)}) \) denotes the cipher text corresponding to the input plain text \( X^{(0)} \oplus A X^{(0)} \). We can identify total 232-bit sub keys, \( R K'_{30} \), \( (I K'_{13})^{\delta_2} \), \( R K'_{42}, I K'_{21}, R K'_{42}, I K'_{22}, R K'_{40} \) and \( I K'_{23} \) by solving these attack equations by an exhaustive search.

Because (8) is 8 sets (bits) of system of Boolean equations, it is satisfied with probability \( 2^{-8} \) even if the estimated sub keys are false. There are \( 2^{232} \) candidates of sub keys since its total bit size is 232. Therefore we need to solve 30 (\( 232/8+1 \)) sets of (8) with different \( X^{(0)} \) in order to identify the true sub key where the probability that a false sub key survives is \( 2^{-8} \). Because we have to compute the 8th-order differential to prepare one set of (8), the number of chosen plain texts to prepare 30 different sets of (8) is given by D as follows:

\[
D = 30 \times 2^8 \approx 2^{12.9} \tag{18}
\]

Next we study the number of times of data encryptions required to solve 30 different sets of (8). If we solve the first set of (8) for all \( 2^{232} \) candidates of sub keys, the number of candidates is reduced to \( 2^{224} \). Then we solve the second set of (8) for the remaining \( 2^{224} \) candidates, its number is reduced to \( 2^{216} \). By solving 30 different sets of (8), the last remaining key will be the true key. \( 2^8 \) times of S-box operation at \( f_6 \) in the 8th round is carried out to check the correctness of one candidate sub key. Because 12 rounds of data-mixing part include 80 S-boxes, the total number of \( 2^{38} \) operations in (8) times 1/80 corresponds to the total number (T) of times of the data encryptions required for this attack as follows:

\[
T = 2^{232} \times 2^8 \times 2^{48} \times \frac{1}{80} \approx 2^{233.7} \tag{19}
\]

V. CONCLUSIONS

We have investigated the new synchronous 8th-order differential attack on 12 rounds of HyRAL as part of security evaluation. We found the new synchronous 8th-order differential characteristics at the 8th round of HyRAL. We equivalently modified \( f_6 \) at the 8th round and derived the attack equations. As a result, we showed that 12 rounds of HyERAL can be attacked with \( 2^{12.9} \) blocks of chosen plain text and \( 2^{233.7} \) times of the data encryption by exploiting the new synchronous 8th-order differential characteristics we found. However full-rounds of HyERAL is secure against our attack because the number of rounds is actually 32.

REFERENCES


Toshinobu Kaneko received the B.E., M.E., and Ph.D. degrees all in Electrical Engineering from the University of Tokyo, in 1971, 1973, and 1976, respectively. In 1976, he joined the faculty of Science and Technology, Tokyo University of Science, and since then, as a faculty member, he has been engaged in education and research in the fields of coding theory and information security. Currently, he is a Professor of Department of Electrical Engineering of the university. Prof. Kaneko is a member of CRYPTREC and served as a chairman of Symmetric-Key Cryptography subcommittee in 2001–2003. Prof. Kaneko is a member of IEICE, IEEJ, IPSJ, and IEEE.

Seiji Fukushima received the B.S., M.S., and Ph.D. degrees in electrical engineering from Kyushu University in 1984, 1986, and 1993, respectively. He is currently a Professor at the Dept. of Electrical and Electronics Engineering, Kagoshima University. His research interests include photonics/radio hybrid communication systems and their related devices. Prof. Fukushima is a member of IEICE, IEEE/Photonic Society, Japan Society of Applied Physics, Japanese Liquid Crystal Society, and Optical Society of America.

Tomohiro Hachino received the B.S., M.S., and Dr. Eng. degrees in electrical engineering from Kyushu Institute of Technology in 1991, 1993, and 1996, respectively. He is currently an Associate Professor at the Dept. of Electrical and Electronics Engineering, Kagoshima University. His research interests include nonlinear control and identification, signal processing, and evolutionary computation. Dr. Hachino is a member of IEEJ, SICE, and ISCIE.