第33卷 第19期 计 算 机 工 程 2007年10月
Vol.33 No.19 Computer Engineering October 2007
·安全技术· 文章编号：1000—3428(2007)19—0164—04 文献标识码：A 中图分类号：TN918

复合混沌伪随机序列加密算法的破译

张 斌，金晨辉
(信息工程大学电子技术学院，郑州 450004)

摘 要：对复合混沌伪随机序列加密算法(CCPRSEA)做了深入分析，通过对该算法中素域上线性同余变换的分析，分离出混沌序列，利用混沌映射自身的信息泄漏规律，给出基于吻合度分布规律和函数中值定理的分割攻击方法，在 *base*=10, *m*=3 的情况下，证明了破译算法的成功率为 0.982 7，计算复杂性为 2^{40}。实验表明，在主频为 2.5GHz 的 Pentium 4 PC 上，求出其全部密钥的整个攻击时间只需 8h35min，因此，CCPRSEA 是不安全的。

关键词：混沌密码；序列密码；分割攻击；已知明文攻击；logistics 映射；线性同余变换

Breaking Method for Composite Chaotic Pseudo-Random Sequence Encryption Algorithm

ZHANG Bin, JIN Chen-hui
(Institute of Electronic Technology, University of Information Engineering, Zhengzhou 450004)

【Abstract】 This paper analyzes the composite chaotic pseudo-random sequence encryption algorithm(CCPRSEA). By analyzing the linear congruent algorithm on a prime field of CCPRSEA, it gains chaotic sequence and gives a tally-degree-based divide-and-conquer attack algorithm by using the information leak of the chaotic map itself. When *base*=10 and *m*=3, it proves that the success rate of the attack algorithm is 0.982 7 and the complexity is 2^{40}. Experimental results indicate that it just needs 8 hours and 35 minutes for attacking all keys of CCPRSEA in 2.5GHz of Pentium 4 PC. So CCPRSEA is insecure.

【Key words】 chaotic cipher; stream cipher; divide-and-conquer attack; known plaintexts attack; logistics mapping; linear congruent algorithm

混沌密码作为当前信息安全领域研究的热点之一得到了广泛关注，出现了很多混沌密码算法[1~3]及一些分析成果[4~6]。当迭代次数足够多时，混沌序列对初值具有极端敏感性。但就其混沌序列的前几个值而言，对初值的变化却不够敏感。这种不敏感性成为本文攻击混沌密码算法的理论依据。

复合混沌伪随机序列加密算法(composite chaotic pseudo-random sequence encryption algorithm, CCPRSEA)是文献[1]提出的基于混沌映射和素域上线性同余变换的复合设计的压缩视频加密算法。本文发现该算法是不安全的，给出了破译该加密算法的攻击算法。

1 CCPRSEA 算法介绍

CCPRSEA 算法包括 2 个编码环节：

(1)logistics 映射：

$$g_\beta(x)=(\beta+1)(1+1/\beta)^\beta x(1-x)^\beta$$

其中，$1\le\beta\le4$；$0<x<1$。

(2)素域 $Z/(2^{31}-1)$ 上的线性同余变换：

$$\varphi(x)=(ax+b)\bmod(2^{31}-1)$$

其中，mod 为取模运算(下同)。

CCPRSEA 算法的密钥是双精度实数(52b) $\beta,x,\Delta\beta,\Delta x$ 和正整数 $base,m$。本文认为双精度实数的有效位数应为 54b，下文的攻击算法均是对 54b 密钥进行的攻击。CCPRSEA 算法如下：

(1)用 $\beta_j=\beta+j\times\Delta\beta, x_0=x+j\times\Delta x$ 初始化 logistics 映射 $g_\beta(x)$ 后，将其迭代 $(base+m\times i)$ 次，$0\le i\le7$，得到状态 $x[i]$。即若 $x_n=g_{\beta_j}(x_{n-1})$，则 $x[i]=x_{base+m\times i}$。依次取 $x[i]$ 的高 32b 得到 8 个 32b 整数，记为 $X[0],X[1],\cdots,X[7]$。这里 $x[i]$ 的高 32b 是从 $x[i]$ 的整数位取起(下同)，显然 $x[i]$ 的最高比特一定为 0，即 $X[i]=\lfloor 2^{31}x[i]\rfloor$，$\lfloor\cdot\rfloor$ 为下取整函数(下同)。

(2)对于 $i:0\le i\le7$，令 $a=X[i\bmod8]$，$b=X[(i+1)\bmod8]$，$S_0=X[(i+2)\bmod8]$，根据 $S_{2k-1}=(aS_{2k-2}+b)\bmod(2^{31}-1)$ 和 $S_{2k}=(bS_{2k-1}+a)\bmod(2^{31}-1)$，$k\ge1$，得到第 i 条 $32\times n$ 比特序列 $S_1S_2,\cdots,S_n$，记该序列为 $L[i]$。

(3)将 8 条序列 $L[0],L[1],\cdots,L[7]$ 连接成 $8\times32\times n$ 比特的乱数序列与第 j 个 GOP 中相应的数据块(文献[1]中将加密复杂度分为 3 个层次，各层次规定了各自的数据块)逐比特异或得到密文。

2 CCPRSEA 算法的破译

定理 1 $0\le i\le7$，若已知序列 $L[i]$ 中任意连续的 5 个 32b，则 $L[i]$ 的整个序列均可得到。

证明 对 $k\ge1$，有

$$S_{2k+1}=(aS_{2k}+b)\bmod(2^{31}-1)=(abS_{2k-1}+a^2+b)\bmod(2^{31}-1)$$

$$S_{2k}=(bS_{2k-1}+a)\bmod(2^{31}-1)=(abS_{2k-2}+b^2+a)\bmod(2^{31}-1)$$

则有

基金项目：河南省杰出青年科学基金资助项目(0312001800)
作者简介：张 斌(1982－)，男，硕士研究生，主研方向：密码学；金晨辉，教授、博士生导师
收稿日期：2006-10-30 **E-mail：**dzjszhangbin@126.com

$$\begin{cases}(S_{2k+1}-S_{2k})=[ab(S_{2k-1}-S_{2k-2})+a^2-b^2-a+b]\bmod(2^{31}-1)\\(S_{2k+2}-S_{2k+1})=[ab(S_{2k}-S_{2k-1})-(a^2-b^2-a+b)]\bmod(2^{31}-1)\end{cases}$$

令：

$$c=ab\bmod(2^{31}-1),\quad d=(a^2-b^2-a+b)\bmod(2^{31}-1)$$

则上式变为

$$\begin{cases}(S_{2k+1}-S_{2k})=[c(S_{2k-1}-S_{2k-2})+d]\bmod(2^{31}-1)\\(S_{2k+2}-S_{2k+1})=[c(S_{2k}-S_{2k-1})-d]\bmod(2^{31}-1)\end{cases}$$

解该方程组可得

$$\begin{cases}c=(S_{2k}-S_{2k-2})^{-1}(S_{2k+2}-S_{2k})\bmod(2^{31}-1)\\d=[S_{2k+1}-S_{2k}-(S_{2k}-S_{2k-2})^{-1}(S_{2k+2}-S_{2k})(S_{2k-1}-S_{2k-2})]\bmod(2^{31}-1)\end{cases}$$

从而有

$$\begin{cases}S_{2k+1}=[S_{2k}+c(S_{2k-1}-S_{2k-2})+d]\bmod(2^{31}-1)\\S_{2k+2}=[S_{2k+1}+c(S_{2k}-S_{2k-1})-d]\bmod(2^{31}-1)\end{cases}\tag{1}$$

$$\begin{cases}S_{2k-1}=[S_{2k}-c^{-1}(S_{2k+2}-S_{2k+1}+d)]\bmod(2^{31}-1)\\S_{2k-2}=[S_{2k-1}-c^{-1}(S_{2k+1}-S_{2k}-d)]\bmod(2^{31}-1)\end{cases}\tag{2}$$

因此，当已知序列 $L[i]$ 中任意连续 5 个 32b 字 $S_{2k-2},S_{2k-1},S_{2k},S_{2k+1},S_{2k+2}$ 时，就可由式(1)和式(2)递归得到 $L[i]$ 的全部时刻的序列。■

由定理 1 可知，可由序列 $L[i]$ 求出初态：

$$S_0=X[(i+2)\bmod 8]\bmod(2^{31}-1)$$

进而可求出 $X[2]',X[3]',\cdots,X[7]',X[0]',X[1]'$，其中，$X[i]'=X[i]\bmod(2^{31}-1)$。

定理 2 设 $X[i]'=X[i]\bmod(2^{31}-1)$，则当 $X[i]'\neq 0$ 时，$X[i]=X[i]'$；当 $X[i]'=0$ 时，$X[i]\in\{0,2^{31}-1\}$。

证明 由 $0\le X[i]\le 2^{31}-1$ 可知 $X[i]'=0$ 等价于 $X[i]\in\{0,2^{31}-1\}$，当 $X[i]'\neq 0$ 时，$X[i]'=X[i]\bmod(2^{31}-1)=X[i]$，即 $X[i]=X[i]'$。■

因此，在已知明文攻击条件下，可由已知的乱数求出序列 $L[0],L[1],\cdots,L[7]$，进而求出这些序列的初态 $X[2]',X[3]',\cdots,X[7]',X[0]',X[1]'$。再利用定理 2 就可剥离线性同余变换求出混沌序列 $X[0],X[1],\cdots,X[7]$。

下面给出对 Logistics 混沌映射的分析。

定理 3 设函数：

$$g_\beta(x)=(\beta+1)(1+1/\beta)^\beta x(1-x)^\beta,\quad 1\le\beta\le 4,\quad 0<x<1$$

则有

$$\left|\frac{\partial}{\partial x}g_\beta(x)\right|<\frac{5^5}{4^3}\approx 48.8,\quad \left|\frac{\partial}{\partial\beta}g_\beta(x)\right|<1+\frac{5^5}{4^4}\times\frac{1}{\mathrm{e}}\approx 5.5$$

由数学分析知识易证明定理 3，这里不再给出。

推论 设函数 $g_\beta(x)=(\beta+1)(1+1/\beta)^\beta x(1-x)^\beta$，$1\le\beta,\beta_0\le 4$，$0<x,x_0<1$，则有

$$\left|g_\beta(x)-g_{\beta_0}(x_0)\right|<48.8\,|x-x_0|+5.5\,|\beta-\beta_0|$$

证明 由中值定理知，存在 ξ_β 和 η_{x_0}，使得

$$g_\beta(x)-g_\beta(x_0)=\frac{\partial}{\partial x}g_\beta(x)\Big|_{x=\xi_\beta}\cdot|x-x_0|$$

$$g_\beta(x_0)-g_{\beta_0}(x_0)=\frac{\partial}{\partial\beta}g_\beta(x_0)\Big|_{\beta=\eta_{x_0}}\cdot|\beta-\beta_0|$$

由定理 3 可知：

$$\left|g_\beta(x)-g_{\beta_0}(x_0)\right|\le\left|g_\beta(x)-g_\beta(x_0)\right|+\left|g_\beta(x_0)-g_{\beta_0}(x_0)\right|<48.8\,|x-x_0|+5.5\,|\beta-\beta_0|$$

■

定理 3 的推论说明，Logistics 混沌映射的信息泄漏规律：输入及参数的低位变化对输出的高位影响不大。因此，本文引入以下定义来刻画该信息泄漏规律。

定义 设 $G_\beta^{(1)}(x)=g_\beta(x)$，$G_\beta^{(m)}(x)=g_\beta(G_\beta^{(m-1)}(x))$，则称 $G_\beta^{(m)}(x)$ 为 $g_\beta(x)$ 的 m 次迭代。两对参数 (β',x') 和 (β,x)，如果 $G_{\beta'}^{(m)}(x')$ 与 $G_\beta^{(m)}(x)$ 的高 tb 全部相同但第$(t+1)$b 不同，则称 t 为 (β',x') 与 (β,x) 的吻合度。设 x 和 β 均是 54b 的双精度小数，$x^{(v)}$ 和 $\beta^{(u)}$ 是分别将 x 和 β 的高 v b 和高 u b 保持不变，其他比特全部置 0 得到的双精度小数，则 $(\beta^{(u)},x^{(v)})$ 与 (β,x) 的吻合度记为 $t_{u,v}$。

因此，Logistics 混沌映射的信息泄漏规律表现为 $t_{u,v}$ 偏大，不服从几何分布。

由于 *base* 和 m 越大，加密速度越慢，因此 *base* 和 m 不可能很大，通过穷举就可确定，在此基础上对参数 β_j 及初态 x_0 进行攻击。在下面的分析中以 *base*=10, m=3 为例，给出具体的攻击算法，对于 *base* 及 m 的其他取值情况，可得出类似的攻击算法。本文先利用分割攻击方法求出 β_j 和 $x[0]$，利用逆推攻击方法由 $x[0]$ 求出 x_0，再对另一个 GOP 攻击最终得到 $\Delta\beta$ 和 Δx，完成对 CCPRSEA 算法的破译。

由定理 1 及定理 2，可得序列 $X[0],X[1],\cdots,X[7]$，对 $0\le i\le 7$，令 $X[i]:=X[i]/2^{31}$，因此，得到了混沌映射 $g_{\beta_j}(x)$ 的 m 次迭代的 7 个输入-输出对的高 32b。

要从理论上精确分析 $t_{u,v}$ 的分布规律是困难的，因此，本文利用模拟实验方法获取 $t_{u,v}$ 的分布规律。随机选取了 1 万组 (β,x)，针对 $(u,v)=(16,32),(32,32),(40,40)$ 这 3 种情形，得到的吻合度 $t_{u,v}$ 的分布律如下：

$t_{16,32}$	≤7	8	9	10	11	12	13	14	15	≥16
个数	2	13	10	29	49	121	225	461	868	8 222

$t_{32,32}$	≤19	20	21	22	23	24	25	26	27	28	29	30	31	32
个数	4	8	10	23	40	81	154	322	548	1 141	1 656	1 525	1 189	3 299

$t_{40,40}$	24	25	26	27	28	29	30	31	32
个数	2	1	1	5	13	22	41	88	9 827

因此，$p(t_{16,32}\ge 12)=0.9897$，$p(t_{32,32}\ge 25)=0.9834$，$p(t_{40,40}\ge 32)=0.9827$。

当 (β',x') 不是 $(\beta^{(u)},x^{(v)})$ 时，可认为 $G_{\beta'}^{(m)}(x')$ 与 $G_\beta^{(m)}(x)$ 近似独立，因此，$G_{\beta'}^{(m)}(x')$ 与 $G_\beta^{(m)}(x)$ 的高 sb 全部相等的概率近似为 $p(t=s)=2^{-(s-1)}$（由于 $G_{\beta'}^{(m)}(x'),G_\beta^{(m)}(x)\in(0,1)$，其整数比特位为 0，因此其最高比特均为 0）。据此就可将 $(\beta^{(u)},x^{(v)})$ 与随机的 (β',x') 区分开。

下面给出攻击 $x[0]$ 和 β_j 的算法。$x[0]$ 的高 32b 已得到，只须求出 $x[0]$ 的低 22b 及 β 的 54b。本算法将首先结合分割攻击与统计分析方法求出 β_j 的高 16b 和次高 16b，然后再求出 β_j 和 $x[0]$ 的高 40b 中的低 8 未知比特，最后求出 $x[0]$ 和 β_j。

算法 1

(1)令 $s_1=12$，$s_2=25$，$s_3=32$。

(2)攻击 $\beta_j^{(16)}$。对 $\beta_j^{(16)}$ 的每个可能值 β_j' 和 $0\le i\le 6$，计算出 $G_{\beta_j'}^{(m)}(X[i])$，如果它与 $X[i+1]$ 的高 s_1 b 全部相等，则令 $\zeta_i(\beta_j')=1$，否则令 $\zeta_i(\beta_j')=0$。记 $T_{\beta_j'}=\sum_{i=0}^{6}\zeta_i(\beta_j')$，将使 $T_{\beta_j'}$ 达到最大的那些 β_j' 存入 $File_1$ 文件中。记 $File_1$ 中存放的候选值

的个数为 N_1。

(3)攻击 $\beta_j^{(32)}$。逐一读出 $File_1$ 文件中的每个候选值 $\beta_j^{(16)}$，将 $\beta_j^{(16)}$ 作为 $\beta_j^{(32)}$ 的高 16b，并对如此得到的 $\beta_j^{(32)}$ 的每个可能的 β_j' 和 $0 \le i \le 6$，计算出 $G_{\beta_j'}^{(m)}(X[i])$，如果它与 $X[i+1]$ 的高 s_2 b 全部相等，则令 $\zeta_i(\beta_j')=1$，否则令 $\zeta_i(\beta_j')=0$。记 $T_{\beta_j'}=\sum_{i=0}^{6}\zeta_i(\beta_j')$，并将使 $T_{\beta_j'}$ 达到最大的那些 β_j' 存入 $File_2$ 文件中。记 $File_2$ 中存放的候选值的个数为 N_2。

(4)攻击 $\beta_j^{(40)}$ 和 $x[0]^{(40)}$。逐一读出 File2 文件中的每个候选值 $\beta_j^{(32)}$，将 $\beta_j^{(32)}$ 作为 $\beta_j^{(40)}$ 的高 32b，将 $X[0]$ 作为 $x[0]^{(40)}$ 的高 32b，并分别穷举 $\beta_j^{(40)}$ 和 $x[0]^{(40)}$ 的剩余 8 b。对如此得到的 $(\beta_j^{(40)},x[0]^{(40)})$ 的每个可能 (β_j',x')，计算出 $G_{\beta_j'}^{(m)}(x')$。如果它与 $X[1]$ 的高 s_3 b 全部相等，则执行(5)；否则检验下个可能的 $(\beta_j^{(40)},x[0]^{(40)})$。当 $(\beta_j^{(40)},x[0]^{(40)})$ 的所有可能值都检验完毕后仍未找到正确值时，报告算法失败。

(5)攻击 β_j 和 $x[0]$。分别将 β_j' 和 x' 作为 β_j 和 $x[0]$ 的高 40b，并对 β_j 和 $x[0]$ 的低 14b 穷举，利用所得的 $(\beta_j,x[0])$ 的每个可能值 (β_j'',x'') 计算出 $x_1''=G_{\beta_j''}^{(m)}(x'')$ 和 $x_{i+1}''=G_{\beta_j''}^{(m)}(x_i'')$，$1 \le i \le 6$。如果对 $1 \le i \le 7$，诸 x_i'' 的高 32b 与 $X[i]$ 的高 32b 都相等，则判定 (β_j'',x'') 为正确密钥，否则判定 (β_j'',x'') 不是正确密钥，返回(5)检验下个可能的 (β_j'',x'')。当 $(\beta_j^{(40)},x[0]^{(40)})$ 的可能值对应的所有可能的 (β_j'',x'') 都检验完毕后仍未找到正确密钥时，返回(4)，检验下个可能的 $(\beta_j^{(40)},x[0]^{(40)})$。当所有可能的 $(\beta_j^{(40)},x[0]^{(40)})$ 都检验完毕后仍未找到正确密钥时，报告算法失败。

引理 设算法 1 的(2)中 $\beta_j^{(16)}$ 的候选值 β_j' 使 $\zeta_i(\beta_j')=1$ 的概率为 $p_{\beta_j'}$，又设诸 $\zeta_i(\beta_j')$ 相互独立，则(2)的输出 $File_1$ 中包含 $\beta_j^{(16)}$ 的概率为

$$\sum_{i=0}^{7}[C_7^i p_{\beta_j^{(16)}}^i(1-p_{\beta_j^{(16)}})^{7-i}\prod_{\beta_j' \in \{0,1\}^{16}\setminus\beta_j^{(16)}}\sum_{k=0}^{i}C_7^k p_{\beta_j'}^k(1-p_{\beta_j'})^{7-k}]$$

将 $\beta_j^{(16)}$ 改为 $\beta_j^{(32)}$，并将 β_j' 改为 $\beta_j^{(32)}$ 的候选值，将集合 $\{0,1\}^{16}\setminus\beta_j^{(16)}$ 改为 $\{0,1\}^{32}\setminus\beta_j^{(32)}$，类似可得到(3)的输出 $File_2$ 中包含 $\beta_j^{(32)}$ 的概率为

$$\sum_{i=0}^{7}[C_7^i p_{\beta_j^{(32)}}^i(1-p_{\beta_j^{(32)}})^{7-i}\prod_{\beta_j' \in \{0,1\}^{32}\setminus\beta_j^{(32)}}\sum_{k=0}^{i}C_7^k p_{\beta_j'}^k(1-p_{\beta_j'})^{7-k}]$$

证明 $T_{\beta_j'}=i$ 等价于在 $\zeta_0(\beta_j'),\cdots,\zeta_6(\beta_j')$ 中，有 i 个为 1，7-i 个为 0，因此，有

$$p(T_{\beta_j'}=i)=C_7^i p_{\beta_j'}^i(1-p_{\beta_j'})^{7-i}$$

又因算法 1 的(2)求出的 β_j' 中包含 $\beta_j^{(16)}$ 等价于 $T_{\beta_j^{(16)}} \ge \max_{\beta_j':\beta_j' \ne \beta_j^{(16)}} T_{\beta_j'}$，且诸 $T_{\beta_j'}$ 相互独立，所以由算法 1 的(2)求出的 β_j' 中包含 $\beta_j^{(16)}$ 的概率为

$$p(T_{\beta_j^{(16)}} \ge \max_{\beta_j' \in \{0,1\}^{16}\setminus\beta_j^{(16)}} T_{\beta_j'})=\sum_{i=0}^{7}p(T_{\beta_j^{(16)}}=i \text{且} \forall \beta_j' \in \{0,1\}^{16}\setminus\beta_j^{(16)}, \text{有} T_{\beta_j'} \le i)$$

$$=\sum_{i=0}^{7}[p(T_{\beta_j^{(16)}}=i)\prod_{\beta_j' \in \{0,1\}^{16}\setminus\beta_j^{(16)}}p(T_{\beta_j'} \le i)]=\sum_{i=0}^{7}[C_7^i p_{\beta_j^{(16)}}^i(1-p_{\beta_j^{(16)}})^{7-i}$$

$$\prod_{\beta_j' \in \{0,1\}^{16}\setminus\beta_j^{(16)}}\sum_{k=0}^{i}C_7^k p_{\beta_j'}^k(1-p_{\beta_j'})^{7-k}] \quad \blacksquare$$

定理 4 算法 1 成功率约为 0.982 7。

证明 算法 1 的成功率为(2), (3), (4)的成功率之积。由上述分析可知

$$p_{\beta_j^{(16)}}=p(t_{16,32} \ge 12)=0.9897, \quad p_{\beta_j'}=2^{-11}$$

$$p_{\beta_j^{(32)}}=p(t_{32,32} \ge 25)=0.9834, \quad p_{\beta_j'}=2^{-24}$$

由引理 1 容易算出 $p(2) \approx p(3) \approx 1$，而 $p(4)=P(t_{40,40} \ge 32)=0.9827$，因此，算法 1 的成功率近似为 0.982 7。$\blacksquare$

本文对 CCPRSEA 算法利用算法 1 进行了 1 例攻击实验。实验表明算法 1 的平均计算复杂性约为 0.875×2^{40} 次函数 $G_\beta^{(m)}(x)$ 的计算。在主频为 2.5GHz 的 Pentium 4 PC 上，整个攻击时间约为 4h17min。实验情况如下：

参数 $base=10$，m=3，密钥 $x_0=0.60001$，$\beta=1.561$(该密钥是文献[1]中实验所用密钥)，则 β 的十六进制表示为 0xc7ced916872b00，$x[0]$ 的十六进制表示为 0x0000554a f18c80。算法 1 中(2)的计算复杂性为 7×2^{16}，(2)得到 3 个候选值 0xc7ce-0xc7d0，因此，(3)的计算复杂性为 $3 \times 7 \times 2^{16}$，(3)得到 4 个候选值 0xc7ced915-0xc7ced918。(4)中当 β_j 的高 32b 被穷举到正确值 0xc7ced916 时，共检验了 2×2^{16} 个实验值，因此，(4)的计算复杂性为 2×2^{16}，此时得到 512 个实验值，它们都进入了(5)，且在(5)中输出了正确密钥，因此，(5)的计算复杂性为 $512 \times 7 \times 2^{28}$，本实验的计算复杂度为

$$7 \times 2^{16}+3 \times 7 \times 2^{16}+2 \times 2^{16}+512 \times 7 \times 2^{28} \approx 0.875 \times 2^{40}$$

算法 1 求出了 $x[0]$ 和 β_j。下面根据定理 3 的推论利用分割攻击方法求出 $g_{\beta_j}(x_{base-1})=x[0]$ 所有可能的逆元 x_{base-1}，然后求出 $g_{\beta_j}(x_{base-2})=x_{base-1}$ 所有可能的逆元 x_{base-2}，经过 $base$ 次求逆过程，就可求出满足 $G_{\beta_j}^{(base)}(x_0)=x[0]$ 的正确密钥 x_0 及其全部等效密钥。

由定理 3 的推论可知，$|g_{\beta_j}(x)-g_{\beta_j}(x_0)|<48.8|x-x_0|$，因此，当 $|x-x_0|<2^{-r}$ 时，一定有

$$|g_{\beta_j}(x)-g_{\beta_j}(x_0)|<48.8 \times 2^{-r}$$

对于区间 (0, 1) 中随机的 y，当 $r \le 5$ 时，由于 $48.8 \times 2^{-5}>1$，因此 $|y-g_{\beta_j}(x_0)|<48.8 \times 2^{-r}$ 恒成立。但是当 $r \ge 6$，由于 $48.8 \times 2^{-r}<1$，因此 $|y-g_{\beta_j}(x_0)|<48.8 \times 2^{-r}$ 只是以概率 48.8×2^{-r} 成立。

据此，将 54b 的输入 x_{base-1} 分成若干块，对于 $k \le (54-6)/r$，记 $x_{base-1}^{(6+kr)}$ 为 x_{base-1} 的高 $6+kr$ 比特构成的小数。利用算法 2 对 x_{base-1} 进行分割攻击。

算法 2

(1)取 $r=2$，$l=ceil(48/r)=24$。

(2)攻击 $x_{base-1}^{(6+r)}$。对 $x_{base-1}^{(6+r)}$ 的每个可能值 x'，计算出 $g_{\beta_j}(x')$，并将使 $|g_{\beta_j}(x')-x[0]|<48.8 \times 2^{-6-r}$ 的 x' 都作为 $x_{base-1}^{(6+r)}$ 的候选值，并将这些候选值写入文件 $File_1$。

(3)对 k=2 到 $k \le ceil(48/r)-1$，依次执行：攻击 $x_{base-1}^{(6+kr)}$。逐一从文件 $File_{k-1}$ 中读出 $x_{base-1}^{(6+(k-1)r)}$ 的候选值，并将之作为 $x_{base-1}^{(6+kr)}$ 的高 $6+(k-1)r$ b；穷举 $x_{base-1}^{(6+kr)}$ 的剩下的 r b，对如此得到的 $x_{base-1}^{(6+kr)}$ 的每个可能值 x'，计算出 $g_{\beta_j}(x')$，将使

$\left|g_{\beta_j}(x')-x[0]\right|<48.8\times2^{-6-kr}$ 的 x' 都作为 $x_{base-1}^{(6+kr)}$ 的候选值，并将这些候选值写入文件 $File_k$。

(4)攻击 x_{base-1} 的其他比特。逐一从文件 $File_{l-1}$ 中读出 $x_{base-1}^{(6+(l-1)r)}$ 的候选值，并将之作为 x_{base-1} 的高 $6+(l-1)r$ b。然后穷举 x_{base-1} 的剩下的 $48-(l-1)r$ b，对如此得到的 $x_{base-1}^{(54)}$ 的每个可能值 x'，计算出 $g_{\beta_j}(x')$，并将与 $x[0]$ 相等的那些 x' 都作为 x_{base-1} 的候选密钥。

定理 5 算法 2 的成功率为 1，平均计算复杂性约为 $2^{6+r}+49.8(l-1)\times2^r$。特别地，当 r=2 时，算法 2 的平均计算复杂性约为 $4837\approx1.18\times2^{12}$。

证明

(1)算法2中，由于 $x_{base-1}^{(6+kr)}$ 与 x_{base-1} 的高 $6+kr$ 比特相同，因此有 $|x_{base-1}^{(6+kr)}-x_{base-1}|<2^{-6-kr}$，从而由定理 3 的推论知，$|g_{\beta_j}(x_{base-1}^{(6+kr)})-g_{\beta_j}(x_{base-1})|<48.8\times2^{-6-kr}$，即正确的 $x_{base-1}^{(6+kr)}$ 在算法 2 的各步中一定被保留，因此，该攻击算法的成功率为 1。

(2)对于 $x_{base-1}^{(6+kr)}$ 的每个错误假设 x'，可认为 x' 产生的 $g_{\beta_j}(x')$ 在区间(0,1)中是随机的，因此，x' 使 $\left|g_{\beta_j}(x')-x[0]\right|<48.8\times2^{-6-kr}$ 成立的概率 $<48.8\times2^{-6-kr}$，$x_{base-1}^{(6+kr)}$ 的 $2^{6+kr}-1$ 个错误假设中使上述不等式成立的个数 $<48.8\times2^{-6-kr}\times(2^{6+kr}-1)<48.8$。又因 $x_{base-1}^{(6+kr)}$ 一定使上述不等式成立，所以 $x_{base-1}^{(6+kr)}$ 的候选密钥个数 <49.8。因此，算法 2 的计算复杂性为

$$2^{6+r}+49.8(l-1)\times2^r$$

特别地，当 r=2 时，其计算复杂性为 $4837\approx1.18\times2^{12}$。■

定理 6 算法 2 攻击 x_0 的成功率为 1。假设算法 2 平均提供 h 个候选密钥，则攻击 x_0 的平均计算复杂性约为 $4837(h^{base}-1)/(h-1)$，且 x_0 的候选密钥数量平均为 h^{base-1}。

证明 算法 2 的成功率为 1，因此，反复利用算法 2 经过 *base* 次求逆运算，一定可以得到 x_0 及其所有的等效密钥。由 $g_{\beta_j}(x)$ 的一个输出求其全部输入的计算复杂性为 4 837，由一个输出平均可求出 h 个输入，则在 *base* 次求 $g_{\beta_j}(x)$ 的输入后，平均执行 $1+h+h^2+\cdots+h^{base-1}=(h^{base}-1)/(h-1)$ 次算法 2，因此，利用算法 2 攻击 x_0 的平均计算复杂性约为 $4837(h^{base}-1)/(h-1)$。由 $g_{\beta_j}(x)$ 的 H 个输出平均可求出 Hh 个可能输入，因此，x_0 的候选密钥数量平均为 h^{base-1}。■

本文利用算法 2 对 x_0 进行了攻击。在主频为 2.5GHz 的 Pentium 4 PC 上，历时 12s，求出了 x_0 的 8 个候选密钥。分别是：

0x4ccd20afa2f05c, 0x29199f2c0e7248, 0xa9318c23a1dfcc,
0xa9318c23a1dfd0, 0x25b7f2e52298c4, 0x07887c9213aa90,
0x283f1ae039554c, 0xaa6794d09ac7c8

其中，第 1 个是正确密钥；其他 7 个是等效密钥。

当利用 2 个已知的 GOP 及其加密结果攻击完 2 个 j 对应的 β_j 和 x_0 后，就可求出 $(\beta,\Delta\beta,x,\Delta x)$ 的若干候选值。再利用第 3 个已知的 GOP 及其加密结果对这些候选值进行检验，就可将密钥 $(\beta,\Delta\beta,x,\Delta x)$ 唯一确定下来。实验表明，当利用 3 个已知的 GOP 及其加密结果攻击 $(\beta,\Delta\beta,x,\Delta x)$ 时，求出的密钥是唯一的。因此，CCPRSEA 算法的密钥很容易全部求出，在主频为 2.5GHz 的 Pentium 4 PC 机上，求出 $(\beta,\Delta\beta,x,\Delta x)$ 需要的时间约为 8h35min。

3 结论

在整数密钥 *base* 和 m 都不大且已知的情况下给出了对 CCPRSEA 算法的已知明文破译方法。显然，CCPRSEA 算法的信息泄漏量随着 m 的增大而减小，但增大 m 会降低 CCPRSEA 算法的加密速度。由攻击方法知，增大 *base* 不能提高 CCPRSEA 算法的抗攻击能力。此外，该算法中的线性同余变换对于抵抗已知明文攻击没有任何帮助。理论分析和实验结果都表明，攻破 CCPRSEA 算法是很容易的。

参考文献

1 Yuan Chun, Zhong Yuzhuo, Yang Shiqiang. Composite Chaotic Pseudo-random Sequence Encryption Algorithm for Compressed Video[J]. Tsinghua Science and Technology, 2004, 9(2):234-241.

2 Fridrich J. Symmetric Ciphers Based on Two-dimensional Chaotic Maps[J]. International Journal of Bifurcation and Chaos, 1998, 8(6): 1259-1284.

3 Tenny R, Tsimring L S. Additive Mixing Modulation for Public Key Encryption Based on Distributed Dynamics[J]. IEEE Transactions on Circuits and Systems, 2005, 52(3): 672-679.

4 金晨辉. 一个基于混沌的分组密码算法的分析[J]. 中国工程科学, 2001, 3(6): 75-80.

5 金晨辉, 高海英. 对两个基于混沌的序列密码算法的分析[J]. 电子学报, 2004, 32(7): 1066-1070.

6 李树钧. 一类混沌流密码的分析[J]. 电子与信息学报, 2003, 25(4): 473-479.

（上接第 160 页）

参考文献

1 Guo H, Georganas N D. Digital Image Watermarking for Arbitrarily Shaped Objects[C]//Proc. of the 21st Biennial Symp. on Communications, Kingston. 2002-06.

2 黄豫蕾. 感兴趣区域的数字水印方法[J]. 计算机应用与软件, 2004, 21(10): 96-98.

3 易开祥, 石教英. 自适应二维数字水印系统[J]. 中国图象图形学报, 2001, 6(5A): 444-449.

4 Philips W. A Comparison of Four Hybrid Block/Object Image Coders[J]. Signal Processing, 1996, 54(10): 99-102.

5 黄继武, Shi Y Q, 程卫东. DCT 域图像水印: 嵌入对策和算法[J]. 电子学报, 2000, 28(4): 386-389.