第 36 卷　第 6 期　　计　算　机　工　程　　2010 年 3 月
Vol.36　No.6　　Computer Engineering　　March 2010
・安全技术・　文章编号：1000—3428(2010)06—0058—02　文献标识码：A　中图分类号：TN918

针对 CSC 系列流密码算法的区分攻击

张庆贵
(解放军信息工程大学电子技术学院，郑州 450004)

摘　要：研究 CSC-(n,N)序列流密码算法簇的安全性，证明产生的第 1 个密钥字节为 0 的概率约为 2^{-n}~2^{-2n}，利用模拟实验验证其正确性，据此提出对 CSC-(n,N)的区分攻击。该区分攻击只需利用 2^{3n+2} 个密钥产生的第 1 个密钥字就能以 0.84 以上的正确率将 CSC-(n,N)产生的密钥流序列与随机序列进行区分。

关键词：密码分析；CSC-(n,N)流密码；区分攻击

Distinguishing Attack on Stream Cipher Algorithm of CSC Family

ZHANG Qing-gui
(Institute of Electronic Technology, PLA University of Information Engineering, Zhengzhou 450004)

【**Abstract**】This paper analyses the safety of stream cipher algorithm of CSC-(n,N) family. It proves that the first word of the key stream produced by CSC-(n,N) is 0 with a probability 2^{-n}~2^{-2n} approximately which is verified by trials. A distinguishing attack on CSC-(n,N) is presented. In the attack, the key stream of CSC-(n,N) can be distinguished with a random stream with the correct probability 0.84 by its first word produced by 2^{3n+2} keys of CSC-(n,N).

【**Key words**】cryptanalysis; CSC-(n,N) stream cipher; distinguishing attack

1　概述

CSC-(n,N)[1]是基于 RC4 设计理念提出的一种系列流密码算法，其参数 n 和 N 是可变的。CSC-(n,N)算法是 RC4 的一种推广，CSC-(8,1)就是 RC4 流密码算法。RC4 算法是 RSA 数据安全公司开发的密钥长度可变的流密码算法，目前广泛应用于商业密码产品中，包括 Lotus Notes、苹果计算机的 AOCE 和 Oracle 安全 SQL 数据库等。但是，由于对 RC4 算法存在许多现实的攻击方法[2-5]，因此 CSC-(n,N)采取利用 N 个独立的 n 进 n 出 S 盒进行链接的设计思想推广 RC4，并期望这种多 S 盒链接设计技术能够提高算法的安全性。

CSC-(n,N)算法的优点在于其需要的存储空间小，且可针对不同的安全性需求选择相应的密钥长度和安全强度。文献[1]给出了其周期等部分密码学性质，但目前还没有该算法的安全性分析结论。通过大量实验说明，随着级联 S 盒个数 N 的增加，CSC-(n,N)的安全性将有所提高。但是，由于其加密速率将随着级联 S 盒个数 N 的增加而降低，因此级联 S 盒的个数 N 应尽量少。对 CSC-(n,N)系列算法产生的第 1 个密钥字的分布规律进行了研究，证明了它为 0 的概率与理想值 2^{-n} 的偏差约为 2^{-2n}，从而提出了对 CSC-(n,N)的区分攻击，并证明了只需 2^{3n} 个样本就可将它产生的密钥流序列与随机序列区分开来，利用模拟实验验证了上述结果的正确性。

2　CSC-N 算法分析

2.1　CSC 算法描述

CSC-(n,N)是级联流密码(Cascaded Stream Cipher, CSC)的简称，它是一个密钥长度可变、且有 n 和 N 这 2 个可变参数的流密码算法。其中，N 是 S 盒的个数；n 是 S 盒的输入和输出比特数。

CSC-(n,N)由初始化过程和密钥流生成过程组成，初始化过程如下：

输入　$2^n l$ bit 的用户密钥 $k_0,k_1,\cdots,k_{l-1}$，其中，$k_0,k_1,\cdots,k_{l-1}$ 都是 n 比特字

输出　CSC-(n,N)流密码中 N 个 S 盒、记忆 $i_1,i_2,\cdots,i_N$ 和 $j_1,j_2,\cdots,j_N$ 的初始状态

```
For z from 0 to 2^n N –1, do K_z=K_{z mod l};
For m from 1 to N, do
  For z from 0 to 2^n –1, do  S_m[z]=z ;
  Do  j_m=0 ;
For m from 1 to N, do
  For  i_m  from 0 to 2^n – 1, do
    j_m=(j_m+S_m[i_m]+K_{2^n(m-1)+i_m})mod2^n ;
    交换 S_m[i_m] 和 S_m[j_m] ;
  i_m=0 ;  j_m = 0 ;
```

密钥流生成算法过程如下：

输入　CSC-(n,N)流密码当前的内部状态

输出　CSC-(n,N)流密码当前时刻输出的 n bit 密钥字和下个时刻的内部状态

```
Step1 i_1 = (i_1+1) mod 2^n;
Step2 For k from 1 to N, do
  Step2.1   j_k = ( j_k + S_k[i_k])mod2^n ;
  Step2.2   交换 S_k[i_k]和 S_k[j_k];
  Step2.3   t = (S_k[i_k]+S_k[j_k])mod2^n;
  Step2.4   if k < N, do i_{k+1} = S_k[t];
            else 将 S_k[t]作为当前输出的密钥字;
```

2.2　对 CSC-(n,N)算法的区分攻击

下面首先分析 CSC-(n,N)算法输出的第 1 个字节为 0 的概率。发现它与均匀分布时为 0 的概率 2^{-n} 有一定的偏差，因而利用此特点就可构造一个区分器对 CSC-(n,N)进行区分攻击。

基金项目：河南省杰出青年科学基金资助项目(0312001800)
作者简介：张庆贵(1963－)，男，博士研究生，主研方向：密码学
收稿日期：2009-11-10　**E-mail：**zhangqingui@126.com

设 CSC-(n,N)算法第一时刻的输出字节为 key_1，下面分析 $key_1=0$ 的概率。以下记 $+_n$ 为模 2^n 加，$-_n$ 为模 2^n 减。

定理 1 设 $S_1,S_2,\cdots,S_N$ 是 CSC-(n,N)初始化后的 N 个 S 盒，B 是 CSC-(n,N)产生第 1 个密钥字过程的 i_N，$C=S_N[B]$，$D=S_N[C]$，再设 key_1 是 CSC-(n,N)产生的第 1 个密钥字，则有

$$key_1=\begin{cases}C & 若 D=0\\ D & 若 C+_n D=B\\ S_N[C+_n D] & 其他\end{cases}$$

证明 只需分析 CSC-(n,N)算法在产生第 1 个密钥字过程中在 $k=N$ 时各步骤的结果即可。

由于 CSC-(n,N)算法在产生第 1 个密钥字过程中开始执行 $k=N$ 时仍有 $j_N=0$，因此在执行完 Step2.1 后，有

$$j_N=j_N+_n S_N[i_N]=0+_n S_N[i_N]=S_N[B]=C$$

和

$$S_N[j_N]=S_N[C]=D$$

再记执行完 Step2.2 后的 S_N 盒为 $S_N^{'}$，则有

$$S_N^{'}[i_N]=S_N[j_N]=D$$

和

$$S_N^{'}[C]=S_N^{'}[j_N]=S_N[i_N]=C$$

因而执行完 $k=N$ 时的 Step2.3 后，有

$$t=S_N^{'}[i_N]+_n S_N^{'}[j_N]=C+_n D$$

故有

$$key_1=S_N^{'}[S_N^{'}[i_N]+_n S_N^{'}[j_N]]=S_N^{'}[C+_n D]$$

为利用 B,C,D,S_N 对 key_1 的具体取值进行描述，可分 3 种情形进行分析：

情形 1 $C+_n D=j_N$。由 $j_N=C$ 知 $C+_n D=j_N$ 等价于 $D=0$，且有

$$key_1=S_N^{'}[C+_n D]=S_N^{'}[j_N]=C$$

情形 2 $C+_n D=i_N$。由 $i_N=B$ 知 $C+_n D=i_N$ 等价于 $C+_n D=B$，且有

$$key_1=S_N^{'}[C+_n D]=S_N^{'}[i_N]=D$$

情形 3 $C+_n D\neq C$ 且 $C+_n D\neq i_2$。此时，有

$$key_1=S_N^{'}[C+_n D]=S_N[C+_n D]$$

此即证得本定理。

定理 2 题设同定理 1，则有

(1)当 $D=0$ 时，$key_1=0$ 等价于 $S_N[0]=0$；

(2)当 $D\neq 0$ 且 $C+_n D=B$ 时，有 $key_1\neq 0$；

(3)当 $D\neq 0$ 且 $C+_n D\neq B$ 时，$key_1=0$ 等价于 $S_N[C+_n D]=0$。

证明

(1)当 $D=0$ 时，由定理 1 知 $key_1=C$。故 $key_1=0$ 等价于 $C=0$。再由 $D=S_N[C]$ 和 $D=0$ 及 S_N 是双射知 $C=0$ 等价于 $S_N[0]=0$。这说明(1)成立。

(2)当 $D\neq 0$ 且 $C+_n D=B$ 时，由定理 1 知 $key_1=D\neq 0$。这说明(2)成立。

(3)当 $D\neq 0$ 且 $C+_n D\neq B$ 时，由定理 1 知 $key_1=S_N[C+_n D]$。这说明(3)成立。

利用定理 1 和定理 2，可得到 $key_1=0$ 的概率。

定理 3 题设同定理 1，并假设

(1)随机变量 D 在 $Z/(2^n)$ 上服从均匀分布；

(2)事件 $D=0$ 与事件 $S_2[0]=0$ 独立；

(3)事件 $D\notin\{0,B-_n C\}$ 的发生概率为 $1-2^{1-n}$；

(4)事件 $D\notin\{0,B-_n C\}$ 与事件 $S_N[C+_n D]=0$ 独立。则有 $p(key_1=0)=2^{-n}-2^{-2n}$

证明 由全概率公式知

$$\begin{aligned}p(key_1=0)=&p(key_1=0\mid D=0)p(D=0)+p(key_1=0\mid\\ &D\neq 0,C+D=B)\times p(D\neq 0,C+D=B)+p(key_1=0\mid\\ &D\neq 0,C+D\neq B)\times p(D\neq 0,C+D\neq B)=\\ &p(S_N[0]=0\mid D=0)p(D=0)+0\times p(D\neq 0,\\ &C+D=B)+p(S_N[C+D]=0\mid D\neq 0,C+D\neq B)\times\\ &p(D\notin\{0,B-_n C\})=p(S_N[0]=0)p(D=0)+\\ &p(S_N[C+D]=0)\times p(D\notin\{0,D\neq B-_n C\})=\\ &2^{-n}\times 2^{-n}+2^{-n}\times(1-2^{1-n})=2^{-n}-2^{-2n}\end{aligned}$$

定理 3 的假设未必一定成立。例如，$N=2$ 时的 Key_1 不均匀分布，意味着 $N=3$ 时的 $B=i_3$ 不均匀分布，因而也难以保证 $N=3$ 的 D 均匀分布。然而，在定理 3 的假设不成立时，可能会导致更严重的不平衡性。实验结果表明，$Key_1=0$ 的概率的确与 2^{-n} 有严重的偏差。

根据定理 3，CSC-$(8,N)$产生的第 1 个密钥字为 0 的概率为 $2^{-8}\sim 2^{-16}$。经编程检验，上述理论结果与实验结果比较接近。在实验中，针对 N=2,3,4，均做了 4 例实验。在每例实验中，通过随机变动初始密钥 K 的方法，由 2^{32} 个密钥产生了 CSC-$(8, N)$的密钥流序列的第 1 个字节，所得的具体实验结果如表 1 所示，其中的数值是偏差 $p(Key_1=0)-2^{-8}$。

表 1　第 1 个密钥字为 0 的概率

N 值	实验 1	实验 2	实验 3	实验 4
2	$-2^{-15.29}$	$-2^{-15.25}$	$-2^{-15.26}$	$-2^{-15.22}$
3	$-2^{-16.36}$	$-2^{-16.41}$	$-2^{-15.57}$	$-2^{-16.55}$
4	$-2^{-16.38}$	$-2^{-16.17}$	$-2^{-16.25}$	$-2^{-16.13}$

定理 3 说明 CSC-(n,N)产生的第 1 个密钥字为 0 的概率 $p_n=2^{-n}\sim 2^{-2n}$ 与理想的概率 2^{-n} 有明显的偏差，据此就可利用现有方法，构造出对 CSC-(n,N)的区分攻击。

设 D_0 和 D_1 是集合 Ω 上的 2 个概率分布，$p_z^{(i)}$ 是随机变量 ξ 服从分布 D_i 时取 z 点的概率。设 $z_1,z_2,\cdots,z_N$ 是随机变量 ξ 的 N 个样本，$z^{(N)}=(z_1,z_2,\cdots,z_N)$。记

$$N(a\mid z^{(N)})=\#\{1\le i\le N:z_i=a\}$$

是 Ω 中元 a 在 $z_1,z_2,\cdots,z_N$ 中的出现次数。令

$$LLR(z^N)=\sum_{a\in\Omega 且 N(a|z^N)>0}N(a\mid z^N)\log\frac{p_a^{(1)}}{p_a^{(0)}}$$

文献[6]证明了当 $LLR(z^N)\ge 0$ 时判定 ξ 服从分布 D_1，否则，判定 ξ 服从分布 D_0 的区分器是使总错误率达到最小的区分器。此外，文献[6]中的定理还指出，若记 $p_z^{(0)}=p_z^{(1)}\oplus\varepsilon_z$，则当 $N=d\Big/\sum_{a\in\Omega}(\varepsilon_a^2/p_a^{(1)})$ 时，该区分器的总错误率为 $1-\Phi(\sqrt{d}/2)$，其中，$\Phi(t)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t}\mathrm{e}^{-\frac{x^2}{2}}\mathrm{d}x$ 是标准正态分布的分布函数。

特别地，设 $\Omega=Z/(2^n)$，D_1 是 $Z/(2^n)$ 上的均匀分布，则 $p_a^{(1)}\equiv 2^{-n}$，因而当 $N=d/(2^n\varepsilon_0^2)>d\Big/(2^n\sum_{a\in Z/(2^n)}\varepsilon_a^2)$ 时，该区分器的总错误率为 $1-\Phi(\sqrt{d}/2)$。

由于 $\varepsilon_0=|p_0^{(0)}-2^{-n}|=2^{-2n}$，因此只需 $O(2^{2-n}\varepsilon_0^{-2})=2^{3n+2}$ 个样本就可实现对 CSC-(n,N)的区分攻击，从而以高于 $\Phi(1)=0.84$ 的正确率，将 CSC-(n, N)产生的第 1 个密钥字与随机数区分开来。特别地，当 $n=8$时，只需要 2^{26} 个样本即可。

(下转第 162 页)