对 Shannon 算法的线性区分攻击

常亚勤*　　金晨辉
(信息工程大学电子技术学院　郑州　450004)

摘　要：该文基于对 Shannon 算法非线性反馈移存器反馈函数和非线性滤波函数进行线性逼近，得到了优势为 2^{-28} 的 32 个新的区分器，给出了一个对流密码算法 Shannon 的新的线性区分攻击。该区分攻击大约需要 2^{52} 密钥字就能将 Shannon 算法的密钥流序列从随机序列中区分出来。

关键词：序列密码；区分攻击；线性逼近；非线性反馈移存器；Shannon 算法
中图分类号：TN918.1　　**文献标识码**：A　　**文章编号**：1009-5896(2011)0190-04
DOI: 10.3724/SP.J.1146.2009.01626

Linear Distinguishing Attack on Shannon Algorithm

Chang Ya-qin　　Jin Chen-hui
(*Institute of Electronic Technology, the University of Information Engineering, Zhengzhou* 450004, *China*)

Abstract: A new distinguishing attack is presented on Shannon algorithm. The distinguish attack is built by using linear approximations of both the non-linear feedback shift register and the non-linear filtration function, and 32 distinguishers are derived which the bias is 2^{-28}. Therefore, the Shannon algorithm is distinguishable from truly random cipher after observing 2^{52} keystreams words on average.

Key words: Stream ciphers; Distinguishing attack; Linear approximations; Non-linear Feedback Shift Register (NFSR); Shannon algorithm

1　引言

Shannon 算法[1]是一个由 Sober 系列算法的设计者 Hawkes 等人设计的同步流密码，面向软件实现。设计者按照 Estream 工程中对面向软件算法的要求来设计，并且使得该算法的硬件实现也非常好。

2008 年，Hakala 等人[2]给出了一个对 Shannon 算法的线性区分攻击，所需的数据复杂度为 $O(2^{109.004})$，2009 年，Hakala 等人[3]又利用多重线性区分攻击改进了文献[2]的区分攻击，所需的数据复杂度为 $O(2^{102})$，Hassanzadeh 等人[4]利用错误诱导的方法给出了对 Shannon 算法的差错区分攻击，2010 年，Zahra 等人[5]利用 CP 攻击的方法给出了对 Shannon 算法实际的区分攻击，所需数据复杂度为 $O(2^{31})$，计算复杂度为 $O(2^{31})$。

本文利用对 Shannon 算法的非线性反馈移存器反馈函数和非线性滤波函数进行线性逼近，给出了对算法的一个新的线性区分攻击。本文建立的 32 个区分器成立的优势都为 2^{-28}，因此平均大约需要 2^{52} 的密钥字就可将 Shannon 算法的密钥流序列从随机序列中区分出来。

2009-12-22 收到，2010-09-14 改回
*通信作者：常亚勤　Chyq1980@yahoo.cn

2　Shannon 算法简介

Shannon 算法是一个由非线性反馈移存器(NFSR)和非线性滤波函数(NLF)组成的流密码算法。NFSR 由 16 级 32 bit 字的存储器组成，在时刻 t 时，记为 $\sigma_t=(r_t[0],r_t[1],\cdots,r_t[15])$，则 NFSR 从时刻 t 到时刻 $t+1$ 的变换为

(1) $r_{t+1}[i]=r_t[i+1], i=1,\cdots,14$；
(2) $r_{t+1}[15]=f_1(r_t[12]\oplus r_t[13]\oplus \text{Konst})\oplus(r_t[0]<<<1)$；
(3) $r_{t+1}[0]=r_t[1]\oplus f_2(r_{t+1}[2]\oplus r_{t+1}[15])$。

其中 Konst 是个由密钥产生的常数，$f_1(x)=g(g(x,5,7),19,22)$，$f_2(x)=g(g(x,7,22),5,19)$，$g(x,a,b)=x\oplus((x<<<a)\vee(x<<<b))$。

本文中“$<<<$”表示循环左移，“$\vee$”表示按位或运算。

算法在每次状态更新后输出 32 bit 字作为密钥流，具体的输出函数为

$$z_t=\text{NLF}(\sigma_t)=r_{t+1}[8]\oplus r_{t+1}[12]\oplus f_2(r_{t+1}[2]\oplus r_{t+1}[15])$$

算法的初始化算法得到 Konst 以及 σ_0，这与我们的分析关系不大，具体参考文献[1]。

3　NFSR 和 NLF 的分析

首先给出优势的定义。

定义 1[2] 设 $\Pr(A)$ 为一个逼近 A 成立的概率，则优势 $\varepsilon = 2\Pr(A) - 1$。

下面分别给出对算法非线性反馈移存器反馈函数和非线性滤波函数的逼近。

Shannon 算法使用了非线性反馈移存器 NFSR，由 NFSR 的结构，可得下列式子成立。

$$f_1(r_t[12] \oplus r_t[13] \oplus \text{Konst}) \oplus (r_t[0] <<< 1) \oplus r_{t+1}[15] = 0 \tag{1}$$

$$r_t[1] \oplus f_2(r_{t+1}[2] \oplus r_{t+1}[15]) \oplus r_{t+1}[0] = 0 \tag{2}$$

设 $x \in Z/(2^n)$ 且 $x = \sum_{k=0}^{n-1} x_k 2^k$，$x_k \in \{0,1\}$。以下本文均称 x_k 是 x 的第 k 位，并将之表示为 $x_{(k)}$。

由于 $f_1(x)$ 和 $f_2(x)$ 的输入变量都为 32 比特，我们分别穷举其输入变量，得到

对 $\forall i: 0 \le i \le 31$，有 $\Pr(f_{1,(i)}(x)=x_{(i)})=5/8, \Pr(f_{2,(i)}(x) = x_{(i)}) = 5/8$，因此 $f_{1,(i)}(x) \oplus x_{(i)} = 0$ 和 $f_{2,(i)}(x) \oplus x_{(i)} = 0$ 成立的优势都为 $2 \times (5/8) - 1 = 2^{-2}$。

将 $f_{1,(i)}(x) = x_{(i)}$ 式和 $f_{2,(i)}(x) = x_{(i)}$ 分别代入式(1)和式(2)中，得

$$r_t[12]_{(i)} \oplus r_t[13]_{(i)} \oplus \text{Konst}_{(i)} \oplus r_t[0]_{(i-1) \bmod 32} \oplus r_{t+1}[15]_{(i)} = 0 \tag{3}$$

$$r_t[1]_{(i)} \oplus r_{t+1}[2]_{(i)} \oplus r_{t+1}[15]_{(i)} \oplus r_{t+1}[0]_{(i)} = 0 \tag{4}$$

其中式(3)和式(4)的优势都为 2^{-2}，将式(4)代入式(3)中，由堆积引理可得

$$r_t[1]_{(i-1) \bmod 32} \oplus r_{t+1}[2]_{(i-1) \bmod 32} \oplus r_t[12]_{(i)} \oplus r_t[13]_{(i)} \oplus r_{t+1}[15]_{(i-1) \bmod 32} \oplus r_{t+1}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0 \tag{5}$$

的优势为 $2^{-2} \times 2^{-2} = 2^{-4}$。

由于非线性滤波函数为 $z_t = r_{t+1}[8] \oplus r_{t+1}[12] \oplus f_2(r_{t+1}[2] \oplus r_{t+1}[15])$，而由 $f_{2,(i)}(x) = x_{(i)}$ 可得到，对 $\forall i: 0 \le i \le 31$

$$r_{t+1}[2]_{(i)} \oplus r_{t+1}[8]_{(i)} \oplus r_{t+1}[12]_{(i)} \oplus r_{t+1}[15]_{(i)} \oplus z_{t,(i)} = 0 \tag{6}$$

的优势为 2^{-2}。

4　对 Shannon 算法的区分攻击

由以上结论，我们得到密钥流输出字的线性逼近式为对 $\forall i: 0 \le i \le 31$，有式(6)成立：

$$r_{t+1}[2]_{(i)} \oplus r_{t+1}[8]_{(i)} \oplus r_{t+1}[12]_{(i)} \oplus r_{t+1}[15]_{(i)} \oplus z_{t,(i)} = 0$$

考虑时刻 $t, t+6, t+10, t+13$ 时 NFSR 的线性逼近，得到

$$\left.\begin{aligned}
&r_t[1]_{(i-1) \bmod 32} \oplus r_{t+1}[2]_{(i-1) \bmod 32} \oplus r_t[12]_{(i)} \oplus r_t[13]_{(i)} \oplus r_{t+1}[15]_{(i-1) \bmod 32} \oplus r_{t+1}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t+6}[1]_{(i-1) \bmod 32} \oplus r_{t+7}[2]_{(i-1) \bmod 32} \oplus r_{t+6}[12]_{(i)} \oplus r_{t+6}[13]_{(i)} \oplus r_{t+7}[15]_{(i-1) \bmod 32} \oplus r_{t+7}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t+10}[1]_{(i-1) \bmod 32} \oplus r_{t+11}[2]_{(i-1) \bmod 32} \oplus r_{t+10}[12]_{(i)} \oplus r_{t+10}[13]_{(i)} \oplus r_{t+11}[15]_{(i-1) \bmod 32} \oplus r_{t+11}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t+13}[1]_{(i-1) \bmod 32} \oplus r_{t+14}[2]_{(i-1) \bmod 32} \oplus r_{t+13}[12]_{(i)} \oplus r_{t+13}[13]_{(i)} \oplus r_{t+14}[15]_{(i-1) \bmod 32} \oplus r_{t+14}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0
\end{aligned}\right\} \tag{7}$$

再由当 $i = 1, \cdots, 15$ 时，$r_{t+1}[i] = r_t[i+1]$，式(7)可表示如下：

$$\left.\begin{aligned}
&r_{t-1}[2]_{(i-1) \bmod 32} \oplus r_{t+1}[2]_{(i-1) \bmod 32} \oplus r_{t+10}[2]_{(i)} \oplus r_{t+11}[2]_{(i)} \oplus r_{t+14}[2]_{(i-1) \bmod 32} \oplus r_{t+14}[2]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t-1}[8]_{(i-1) \bmod 32} \oplus r_{t+1}[8]_{(i-1) \bmod 32} \oplus r_{t+10}[8]_{(i)} \oplus r_{t+11}[8]_{(i)} \oplus r_{t+14}[8]_{(i-1) \bmod 32} \oplus r_{t+14}[8]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t-1}[12]_{(i-1) \bmod 32} \oplus r_{t+1}[12]_{(i-1) \bmod 32} \oplus r_{t+10}[12]_{(i)} \oplus r_{t+11}[12]_{(i)} \oplus r_{t+14}[12]_{(i-1) \bmod 32} \oplus r_{t+14}[12]_{(i)} \oplus \text{Konst}_{(i)} = 0 \\
&r_{t-1}[15]_{(i-1) \bmod 32} \oplus r_{t+1}[15]_{(i-1) \bmod 32} \oplus r_{t+10}[15]_{(i)} \oplus r_{t+11}[15]_{(i)} \oplus r_{t+14}[15]_{(i-1) \bmod 32} \oplus r_{t+14}[15]_{(i)} \oplus \text{Konst}_{(i)} = 0
\end{aligned}\right\} \tag{8}$$

将式(8)中各等式的左右两边分别相加，可得

$$z_{t-1,(i-1) \bmod 32} \oplus z_{t+1,(i-1) \bmod 32} \oplus z_{t+10,(i)} \oplus z_{t+11,(i)} \oplus z_{t+14,(i-1) \bmod 32} \oplus z_{t+14,(i)} = 0 \tag{9}$$

因此，对任意选定的 i $(0 \le i \le 31)$，式(9)的优势为 $(2^{-4})^4 (2^{-2})^6 = 2^{-28}$。故由式(9)，我们就得到了 32 个优势都为 2^{-28} 的区分器。

下面给出对 Shannon 算法的区分攻击。一般来说，对流密码的区分攻击是要找出 t，并利用第 i 个密钥流输出字至第 $i+t$ 个密钥流输出字，构造出一个不在 $\{0,1\}^m$ 上均匀分布的 m 维二元随机向量 $\boldsymbol{\xi}_i(k, \text{IV})$，并借助于此将密钥流序列与随机序列区分开来，这里 $\boldsymbol{\xi}_i(k, \text{IV})$ 独立同分布。区分方法主要有两类：第 1 类是对固定的 IV 和密钥 k，变动 i，并借助于 $\{\boldsymbol{\xi}_i(k, \text{IV})\}_{i=1}^{\infty}$ 进行区分；第 2 类是对固定 i，变动 (k, IV)，并借助于 $\{\boldsymbol{\xi}_i(k, \text{IV}) : (k, \text{IV}) \in \boldsymbol{\Omega}\}$ 进行区分。具体的区分方法[5,6]是由给定的 N 个样本 $\boldsymbol{a}^{(1)}, \boldsymbol{a}^{(2)}, \cdots, \boldsymbol{a}^{(N)}$ 计算出诸 $p(\boldsymbol{\xi} = a^{(i)})$，从而计算出判据 $T = \sum_{i=1}^{N} \log(2^m p(\boldsymbol{\xi} = a^{(i)}))$。当 $T > 0$ 时，判定密钥流序列不服从均匀分布，否则判定密钥流序列服从均匀分布。

若记 $\varepsilon = 2^m \sum_{a \in \{0,1\}^m} [p(\boldsymbol{\xi}_i(k, \text{IV}) = a - 2^{-m}]^2$，$\boldsymbol{\Phi}$ 是标准正态分布的概率函数。由文献[6，7]的证明可知，当 N 充分大时，将一条与 $\boldsymbol{\xi}_i(k, \text{IV})$ 同分布的序列判定为不服从均匀分布的概率与将一条均匀分布的序列判定为不服从均匀分布的概率之差(即区分优势[8–10])是 $P = 1 - 2\boldsymbol{\Phi}(-\sqrt{N\varepsilon}/2)$，因此，当 $N = \varepsilon^{-1}$ 时，有 $P = 1 - 2\boldsymbol{\Phi}(-0.5) \approx 0.38$；当 $N = 2\varepsilon^{-1}$

时，有 $P\approx 0.52$。这说明在要求区分优势为 0.52 的条件下，第 1 类区分攻击所需密钥流长度是 $2\varepsilon^{-1}$，第 2 类区分攻击则需 $2\varepsilon^{-1}$ 个 (k,IV) 产生的密钥流片段。二者的计算复杂性都是 $N\times C$，这里 C 是计算 $\log(2^m p(\boldsymbol{\xi}=a_i))$ 时所需的计算量。

本文中若利用式(9)中 32 个区分器中的一个，则需要 2^{56} 密钥流输出字就可以 0.52 的区分优势对 Shannon 进行区分攻击。若我们得到的 32 个区分器是互相独立的，则可以同时利用式(9)中得到的 32 个区分器来对 Shannon 算法进行区分攻击。下面分析同时利用 32 个区分器所需的数据量。

利用式(9)中得到的 32 个区分器，使用第 1 类区分攻击对 Shannon 算法进行分析。若设 $w_i=z_{t-1,(i-1)\bmod 32}\oplus z_{t,(i-1)\bmod 32}\oplus z_{t+11,(i)}\oplus z_{t+12,(i)}\oplus z_{t+13,(i-1)\bmod 32}\oplus z_{t+14,(i)}$，$g\in Z/(2^{32})$，$\boldsymbol{g}=(g_{31},g_{30},\cdots,g_0)=(w_{31},w_{30},\cdots,w_0)$，我们首先给出 w_i 和 $\boldsymbol{g}$ 的概率取值及 $\boldsymbol{g}$ 平方和的计算公式。

定理 1 设 z_t 为 Shannon 算法第 t 时刻的密钥流输出字，若设 $w_i=z_{t-1,(i-1)\bmod 32}\oplus z_{t,(i-1)\bmod 32}\oplus z_{t+11,(i)}\oplus z_{t+12,(i)}\oplus z_{t+13,(i-1)\bmod 32}\oplus z_{t+14,(i)}$，则有

$$\Pr(w_0=0)=\Pr(w_1=0)=\cdots=\Pr(w_{31}=0)=\frac{1+2^{-28}}{2}$$

证明 由式(9)易得。 证毕

定理 2 设 z_t 为 Shannon 算法第 t 时刻的密钥流输出字，若设 $w_i=z_{t-1,(i-1)\bmod 32}\oplus z_{t,(i-1)\bmod 32}\oplus z_{t+11,(i)}\oplus z_{t+12,(i)}\oplus z_{t+13,(i-1)\bmod 32}\oplus z_{t+14,(i)}$，$\boldsymbol{g}\in Z/(2^{32})$，$\boldsymbol{g}=(g_{31},g_{30},\cdots,g_0)=(w_{31},w_{30},\cdots,w_0)$，若 $\Pr(w_0=j),\cdots,\Pr(w_{31}=j)(j=0,1)$ 相互独立，则有 $\Pr(\boldsymbol{g}=a)=\left(\frac{1+2^{-28}}{2}\right)^{32-wt(a)}\left(\frac{1-2^{-28}}{2}\right)^{wt(a)}$，其中 $wt(a)$ 表示 a 的汉明重量。

证明 由 $\boldsymbol{g}=(w_{31},w_{30},\cdots,w_0)$，且 $\Pr(w_0=j),\cdots,\Pr(w_{31}=j)$ $(j=0,1)$ 相互独立得

$$\Pr(\boldsymbol{g}=a)=\prod_{i=0}^{31}\Pr(w_i=j),\ j=0,1$$

由定理 1，得 $\prod_{i=0}^{31}\Pr(w_i=j)=\Pr(w_i=0)^{32-wt(a)}\Pr(w_i=1)^{wt(a)}$，故 $\Pr(\boldsymbol{g}=a)=\left(\frac{1+2^{-28}}{2}\right)^{32-wt(a)}\cdot\left(\frac{1-2^{-28}}{2}\right)^{wt(a)}$。 证毕

定理 3 设 z_t 为 Shannon 算法第 t 时刻的密钥流输出字，若设 $w_i=z_{t-2,(i-1)\bmod 32}\oplus z_{t,(i-1)\bmod 32}\oplus z_{t+10,(i)}\oplus z_{t+11,(i)}\oplus z_{t+13,(i-1)\bmod 32}\oplus z_{t+14,(i)}$，$\boldsymbol{g}\in Z/(2^{32})$，$\boldsymbol{g}=(g_{31},g_{30},\cdots,g_0)=(w_{31},w_{30},\cdots,w_0)$，有 $\sum_{a=0}^{2^{32}-1}[\Pr(g=a)]^2=\frac{(1+2^{-56})^{32}}{2^{32}}$。

证明 由定理 2，$\Pr(\boldsymbol{g}=a)=\left(\frac{1+2^{-28}}{2}\right)^{32-wt(a)}\cdot\left(\frac{1-2^{-28}}{2}\right)^{wt(a)}$，则

$$\begin{aligned}\sum_{a=0}^{2^{32}-1}[\Pr(\boldsymbol{g}=a)]^2&=\left(\left(\frac{1+2^{-28}}{2}\right)^2\right)^{32}+C_{32}^{1}\left(\left(\frac{1+2^{-28}}{2}\right)^2\right)^{31}\cdot\left(\frac{1-2^{-28}}{2}\right)^2+\cdots+C_{32}^{31}\left(\frac{1+2^{-28}}{2}\right)^2\cdot\left(\left(\frac{1-2^{-28}}{2}\right)^2\right)^{31}+\left(\left(\frac{1-2^{-28}}{2}\right)^2\right)^{32}\\&=\left(\left(\frac{1+2^{-28}}{2}\right)^2\right)^{2}+\left(\left(\frac{1-2^{-28}}{2}\right)^2\right)^{32}\\&=\frac{(1+2^{-56})^{32}}{2^{32}}\end{aligned}$$

证毕

设随机变量 $\boldsymbol{\chi}=\boldsymbol{g}$，则利用定理 2 和定理 3 可直接得到随机变量 $\boldsymbol{\chi}$ 的概率值及其平方和取值。下面将借助该 32 维二元随机向量 $\boldsymbol{\chi}$ 在 $\{0,1\}^{32}$ 中分布的不平衡性，以 $T=\sum_{i=1}^{N}\log(2^{32}p(\boldsymbol{\chi}=a^{(i)}))$ 为判据，提出对 Shannon 的新的区分攻击方法。这里 $a^{(i)}$ 是已知的由固定的 (k,IV) 产生的时刻 $i+2$ 时的 $\boldsymbol{g}$ 的值。若区分优势为 0.52，所需的密钥流长度为 $N=2\varepsilon^{-1}$。

$$\begin{aligned}\varepsilon&=2^{32}\sum_{a=0}^{2^{32}-1}[\Pr(\boldsymbol{g}=a)-2^{-32}]^2\\&=2^{32}\sum_{a=0}^{2^{32}-1}[\Pr(\boldsymbol{g}=a)-2\times 2^{-32}\Pr(g=a)+2^{-64}]\\&=2^{32}\sum_{a=0}^{2^{32}-1}[\Pr(\boldsymbol{g}=a)]^2-2^{-31}+2^{-32}\\&=2^{32}[2^{-32}(1+2^{-56})^{32}-2^{-32}]\approx 32\times 2^{-56}=2^{-51}\end{aligned}$$

因此，只需由固定 (k,IV) 产生度为长 $2\times 2^{51}=2^{52}$ 密钥流输出字就可以 0.52 的区分优势对 Shannon 进行区分攻击。

说明 上述对区分攻击数据的估计是按照 $\boldsymbol{g}=(g_{31},g_{30},\cdots,g_0)$ 的各分量之间是相互独立的，但是 $\boldsymbol{g}$ 的各分量之间并不一定独立，若它们不独立，则数据量的计算非常困难，我们可按照独立来近似估计。

5 结束语

本文主要提出了对 Shannon 算法的一种新的线性区分攻击方法。我们首先对 Shannon 算法的非线性反馈移存器反馈函数和非线性函数进行线性逼

近，建立了 32 个优势为 2^{-28} 的区分器，进而利用 32 维二元随机向量在 $\{0,1\}^{32}$ 中分布的不平衡性，给出了对算法的区分攻击，攻击所需的数据为 2^{52} 的密钥字，区分优势为 0.52。

参 考 文 献

[1] Hawkes P and McDonald C, *et al*.. Design and primitive specification for Shannon stream cipher[EB]. http://eprint.iacr. org/2007/044, 2007.

[2] Hakala R M and Nyberg K. Linear Distinguishing attack on Shanaon[C]. ACISP 2008, 2008, LNCS 5107: 297–305.

[3] Hakala R M and Nyberg K. A multidimensional linear distinguish attack on Shanaon[J]. *International Journal of Applied Cryptography*, 2009, 1(3): 161–169.

[4] Hassanazadeh M M and Parker M G, *et al*.. Differential distinguishing attack on Shannon based fault analysis[C]. International Symposium on Telecommunications 2008: 671–676.

[5] Zahra A, Javad M and Risto M, *et al*.. A practical distinguisher for the Shannon cipher[J]. *Journal of Systems and Software*, 2010, 83(4): 543–547.

[6] Crowley P. Improved cryptanalysis of Py[R]. ECRYT Stream Cipher Project, Report 2006/010, 2006.

[7] Baigneres T, Junod P, and Vandenay S. How far can we go beyond linear cryptanalysis[C]. In Advances in Cryptology -Asiacrypt 2004, LNCS 3329: 432–450.

[8] 陈士伟, 金晨辉. 模 2 加整体逼近二元和三元模 2^n 加的噪声函数分析[J]. 电子与信息学报, 2008, 30(6): 1445–1449.
Chen S W and Jin C H. Analysis of noise functions of macrocosm approximation of binary addition and tripe addition modulo 2^n with XOR [J]. *Journal of Electronics and Information Technology*, 2008, 30(6): 1445–1449.

[9] 张龙, 吴文玲, 温巧燕. Mod 2^n 加运算与 F_2 上异或运算差值的概率分布和递推公式[J]. 北京邮电大学学报, 2007, 30(1): 85–89.
Zhang L,Wu W L, and Wen Q Y. Probability distribution and recursive formula of difference between mod 2^n sum and XOR over F_2[J]. *Journal of Beijing University of Posts and Telecommunications*, 2007, 30(1): 85–89.

[10] Cho Joo-yeon and Pieprzyk J. An improved distinguisher for dragon[R]. ESTREAM, ECRYPT Stream Cipher Project, Report 2007/002, 2007.

常亚勤：　女，1980 年生，博士生，研究方向为密码学.

金晨辉：　男，1965 年生，教授，博士生导师，研究领域为密码学与信息安全.