简评美国公布的15个AES候选算法*

吴文玲 冯登国 卿斯汉

(中国科学院软件研究所 北京 100080)
(中国科学院信息安全技术工程研究中心 北京 100080)

摘要 文章对美国国家标准和技术研究所(NIST)最近公布的15个AES候选算法的基本设计思想作了简要介绍,同时也介绍了对这些算法的最新分析结果.

关键词 AES(advanced encryption standard),DES(data encryption standard),加密,解密,密钥.
中图法分类号 TP391

从各方面来看,DES(data encryption standard)已走到了它生命的尽头.最近的秘密密钥挑战赛已证明DES的56-比特密钥太短.虽然三重DES可以解决密钥长度的问题,但是DES的设计主要针对硬件实现.而今,在许多领域,需要针对软件实现相对有效的算法.鉴于此,1997年4月15日,美国国家标准和技术研究所(NIST)发起征集AES(advanced encryption standard)算法的活动[1],并成立了AES工作组.其目的是为了确定一个非保密的、公开披露的、全球免费使用的加密算法,用于保护下一世纪政府的敏感信息,也希望能够成为秘密和公开部门的数据加密标准(DES).1997年9月12日,NIST在联邦登记处(FR)公布了征集AES候选算法的通告.AES的基本要求是比三重DES快,且至少和三重DES一样安全,分组长度是128比特,密钥长度为128,192和256比特.1998年8月20日,NIST召开了第1次AES候选会议,并公布了15个AES候选者.目前工作正处在评论阶段.除了NIST对这些候选者做大量的分析和测试外,也欢迎每个对此感兴趣的部门和个人提出自己的看法和意见.1999年3月22日,NIST将举行第2次AES候选会议,公开15个候选算法的讨论结果.关于这些算法的讨论将于1999年4月15日结束,到时会从这15个算法中选出5个.最终在这5个算法中遴选出一个算法作为AES,预计于2001年出台.为了让更多的国内读者了解这些候选算法的近况,本文简要介绍这些算法的基本设计思想和最新分析结果.

1 15个AES候选算法

1.1 CAST-256算法

CAST-256算法是加拿大的Carlisle Adams提交的一个候选算法.它是在已有的CAST-128算法[2]的基础上设计的,该算法采用的总体结构是推广的Feistel网络,此结构最具吸引力的特点是:它能够直接重用来自(传统的)Feistel网络的轮函数.众所周知,增加轮函数的规模一般包括增加它的S-盒的规模或增加它的扩散层的规模.而增加S-盒的规模及增加扩散层的规模往往是比较困难的.因此,推广的Feistel网络无疑会给设计者带来许多方便.

CAST-256的轮函数具有如下特点:

(1) S-盒的设计保证它具有较高的非线性性、较低的差分均匀性、良好的高阶严格雪崩特性以及良好的高阶(输出)比特独立准则.

---

* 本文研究得到国家自然科学基金资助.作者吴文玲,女,1966年生,博士后,主要研究领域为密码理论和应用数学.冯登国,1965年生,研究员,主要研究领域为信息安全理论和技术.卿斯汉,1939年生,研究员,博士生导师,主要研究领域为信息安全理论和技术.

本文通讯联系人:卿斯汉,北京 100080,中国科学院软件研究所

本文1999-01-19收到原稿,1999-02-12收到修改稿

(2) 采用"掩蔽"密钥和"循环移位"密钥,确保每圈中密钥熵高于数据熵,并且可使差分及线性密码分析变得更加困难.

(3) 来自不同群运算的混合似乎可以有效地减少轮差分特征的概率,同样也可以减少高阶差分攻击的可能性.

已有的分析结果显示,CAST-256对现有的分析方法是安全的.用线性密码分析攻击[3]它所需的已知明文大约为2^{174},用差分密码分析攻击[4]40轮CAST-256所需的选择明文大约为2^{140}[5],高阶差分密码分析[6]对5轮以上的CAST-256就无能为力了[7].

1.2 CRYPTON算法

CRYPTON算法是韩国的Chae Hoon Lim提交的一个候选算法.该算法采用的结构是代替-置换网络,它的设计思想和SQUARE算法的一样,把明文输入到一个4×4的字节矩阵,进行字节矩阵的并行字节代替处理,列方式的比特置换,列到行的转置,然后执行密钥加法,实现对128比特数据的加密.

在CRYPTON算法中主要使用了以下几种变换:(1) 非线性替换γ,(2) 线性变换π和τ,(3) 密钥加变换σ.轮变换由γ,π,τ和σ共4种变换复合而成,且奇数轮和偶数轮的轮变换稍有不同,其中采取的一些技巧主要是为了保证加解密相同.非线性替换γ是为了实现"混淆"而设计的,它使用了两个8×8的S-盒S_0和S_1.这两个S-盒是从3个4×4的S-盒使用一个3轮Feistel结构构作出来的,并满足:对任何8比特长的数x,有$S_0(S_1(x))=S_1(S_0(x))=x$.线性变换$\pi$和$\tau$是为了实现"扩散"而设计的,且它的扩散特性没有SQUARE算法中所用的扩散层的性质好.

已有的分析结果显示,CRYPTON对现有的分析方法是安全的.用线性密码分析攻击它所需的已知明文不少于2^{128}.8轮CRYPTON的最佳差分特征的概率不会比2^{-160}大.高阶差分密码分析及多重线性密码分析对CRYPTON的安全性没有造成影响.

1.3 DEAL算法

DEAL算法是加拿大和挪威的Richard Outerbridge等人提交的候选算法.它采用的总体结构是一个6轮的Feistel网络,轮函数是DES算法.它的设计思想就是重用DES.由于所选用的轮函数非常复杂,考虑到速度问题,因此,DEAL的轮数不能多,建议用6轮或8轮.对于6轮的Feistel密码,可以找到一个概率为0的5轮差分,利用此差分攻击DEAL需2^{70}个选择明文、2^{121}次加密及2^{84}个存储字.

1.4 DFC算法

DFC算法是基于Vauenay[8,9]的抗相关攻击技术设计的.它采用的是8轮的Feistel结构.函数F定义如下:

$$F_{(a,b)}=CP(((aX+b) \bmod (2^{64}+13)) \bmod 2^{64}).$$

其中X是64比特输入,$K_i=(a,b)$是子密钥.

CP置换定义如下:

$$CP(Y)=(((Y_r \oplus RT(trunc_6(Y_l)))\,|\,(Y_r \oplus KC))+KD) \bmod 2^{64}.$$

我们已证明CP置换的密码学特性非常不好.因此,DFC的安全性完全依赖于子密钥的嵌入方式.

1.5 E2算法

E2算法是日本NTT公司提交的一个候选算法,它采用的总体结构是Feistel网络.Feistel网络有一点不足之处,即只有一半明文直接输入到第1轮的轮函数,同时,只有一半密文直接输入到最后一轮的轮函数.这意味着在选择明文攻击时,攻击者可以控制第1轮和最后一轮的轮函数的输入;在已知明文攻击时,攻击者可以知道第1轮和最后一轮的轮函数的输入.根据实际经验,攻击者总是可以利用这种缺陷去攻击Feistel密码第1轮和最后一轮的子密钥.为此,E2算法添加了初始变换和最终变换,分别为IT函数和FT函数,它们依赖于子密钥,并含有异或运算之外的运算.

轮函数的设计采用了2圈SPN结构.此结构由3层组成,第1层为第1个非线性层,第2层为双射线性层,第3层为第2个非线性层.非线性层由8个并行的S-盒构成,所起的作用是"混淆".线性层是一个线性变换P,所起的作用是"扩散".此种构造轮函数的优点是可以给出轮函数的最佳差分概率和线性偏差的界.

S-盒的选取使得它的差分均匀性和最佳线性偏差分别为 $2^{-4.67}$和 $2^{-3.19}$,次数为7.线性变换 P 的分支数为5.因此,8轮E2算法的最佳差分特征概率和最佳线性逼近偏差分别为 $2^{-93.4}$和 $2^{-44.8}$.12轮E2算法的最佳差分特征概率和最佳线性逼近偏差分别为 2^{-140}和 $2^{-66.7}$.因此可以说,E2算法对现有的分析方法是免疫的.

1.6 FROG算法

FROG算法是一种非正规结构的密码算法,其基本设计思想是通过内部密钥隐藏大多数计算过程,也就是尽可能不给攻击者对实际执行过程有更多的了解,从而挫败任何攻击.FROG的密钥长度是可变的,从5个字节到125个字节.8轮FROG需要内部密钥包含2 304个字节,因此,FROG的密钥编排的工作量比较大.

关于FROG的最新分析结果有:

(1) 对大约 2^{-33}的密钥,用差分密码分析破译FROG所需的选择明文为 2^{58}个.

(2) 对大约 $2^{-31.8}$的密钥,用线性密码分析破译FROG所需的已知明文为 2^{56}个.

(3) 对大约 $2^{-31.8}$的密钥,存在唯密文的线性密码分析破译FROG所需的密文为 2^{64}个.

(4) FROG的解密函数比加密函数弱得多.对解密函数进行差分密码分析,大约对 2^{-33}的密钥所需的选择明文为 2^{36}.

我们断定,FROG算法不会入选AES.

1.7 HPC算法

HPC算法的分组长度和密钥长度都是任意的,0比特或多个比特.HPC有512比特可选的二级密钥,即SPICE,一个主密钥对每个二级密钥值给出一个不同的加密.HPC由5个不同的子密码组成,究竟采用哪一个子密码由所需加密的消息的长度来决定.每个子密码使用自己的密钥扩展表,这些密钥扩展表通过密钥伪随机产生,每个表共有256个64比特字.所有的密钥扩展表都使用同一个算法,只有初始值不同.

HPC的内部使用无符号的64比特字,任何可变的值,比如密钥长度、明文长度或密文长度都以多个64比特字表示,可能后接一个适当调整的零碎字.SPICE由8个字的数组指定.主要使用的操作有模加、模减、模乘、异或和移位.

人们认为HPC设计得非常不好,设计者似乎不懂密码的基本原则,没有任何可供参考的价值,不可能充当AES,我们也赞同这种观点.

1.8 LOKI97算法

LOKI97算法采用的是Feistel结构,它是LOKI系列密码的最新产品.LOKI89由Biham和Shamir[10]进行了分析,结果表明,虽然LOKI89减少几轮变体可能易受差分密码分析的攻击,但全部16轮的LOKI89却经得起差分密码分析的攻击.Tokita Sorimachi和Matsui[11]进行了线性密码分析,发现12轮以上的LOKI91对线性密码分析是安全的.LOKI97选取的S-盒的非线性非常好.轮函数的构作如下:

轮函数 $f:F_2^{64}\times F_2^{64}\to F_2^{64}$,

$$f(A,B)=S_b(P(S_a(E(KP(A,B)))),B).$$

$KP(A,B)$是一个简单的密钥控制置换.E是一个扩展函数,E从64个输入比特中产生一个96比特输出值.S_a由盒 S_1 和盒 S_2 并置构成,$S_a=[S_1,S_2,S_1,S_2,S_2,S_1,S_2,S_1]$.$P$是一个线性变换.$S_b$由盒 S_1 和盒 S_2 并置构成,$S_b=[S_2,S_2,S_1,S_1,S_2,S_2,S_1,S_1]$,其输入的一部分是子密钥比特.

虽然LOKI97是LOKI系列密码的最新产品,轮函数似乎也很复杂,但经分析发现,LOKI97是很脆弱的.用差分密码分析攻击它所需的选择明文为 2^{56}个.对某些密钥,用线性密码分析攻击它所需的已知明文为 2^{50}个.

1.9 MAGENTA算法

MAGENTA算法的核心部分是以快速哈达马变换为基础的,算法的设计思想是应用一些可在软件和硬件中有效实现的简单且透明的技术.MAGENTA采用的是6轮或8轮的Feistel结构.MAGENTA算法的密钥编排太简单,把种子密钥分成64比特的子块,并在加密过程中对称使用.我们认为这是MAGENTA算法的缺陷,利用它,可以给出攻击MAGENTA的方法,此方法所需的选择明文为 2^{64}个.

1.10 MARS算法

MARS算法是IBM公司提供的一个候选算法,它的特点是充分使用非平衡的Feistel网络.为了保证加密

和解密的强度相当,MARS 由结构类似的两部分组成. MARS 的加密算法由 6 部分组成,第 1 部分是密钥加,第 2 部分是不受密钥控制的 8 轮前期混合运算,第 3 部分是密钥控制下的 8 轮前期加密变换,第 4 部分是密钥控制下的 8 轮后期加密变换,第 5 部分是不受密钥控制的 8 轮后期混合运算,第 6 部分是密钥减.

许多密码分析方法对第 1 和最后一轮的处理方法与中间轮不一样,一般都是首先猜测几比特密钥,然后剥去密码的第 1 和最后一轮,再将攻击施加于剩下的轮上. 鉴于此,MARS 使用了前期混合运算和后期混合运算. 对第 1 和最后一轮特殊对待,我们认为这是很有必要的. 但是,MARS 做得过于复杂,共用了 16 轮,这可能会影响其速度.

MARS 的 S-盒是根据公开原理设计的,即采用随机方式生成并检测,使其具有良好的差分和线性特性,这可以排除预制陷门的嫌疑.

E 函数是 MARS 的核心,它的输入是 32 比特的字 X,输出是 3 个 32 比特的字(Y_2,Y_1,Y_0),

$$Y_2=ROL(ROL(X,13)\times k',10),$$

$$Y_1=ROL[(X+k),ROR(Y_2,5)],$$

$$Y_0=ROL(S(X+k)\oplus ROL(Y_2,5)\oplus Y_2,Y_2),$$

S 是 9×32 的盒子,k 和 k' 是子密钥,$ROR(X,n)$表示 X 右旋转 n 比特,$ROL(X,n)$表示 X 左旋转 n 比特.

E 函数通过使用模 2^{32}乘和数据相依旋转来获得安全性. 模 2^{32}乘的主要密码强度在乘积的高次比特,因为这些比特的每一个几乎都以非线性方式依赖做域的所有比特,并且这些比特有极强的差分特性,所以,在 E 函数中用乘积的高次比特作为数据相依旋转的循环次数,使得 MARS 具有良好的抗差分分析能力. 值得注意的是,乘法运算的代阶比较高,它的运算时间是其他运算时间的 2 倍,且硬件实现代阶更高. 而且正是使用了模 2^{32}乘,使得 MARS 有弱密钥.

从现有的分析结果来看,MARS 对现有的密码分析方法是免疫的. 其缺陷是有弱密钥,且速度相对较慢.

1.11 RC6 算法

RC6 算法是在 RC5 算法的基础上设计的. 众所周知,RC5 是一个非常简洁的算法,它的特点是大量使用数据依赖循环. RC6 继承了这些优点. 为了满足 NIST 的要求,即分组长度为 128 比特,RC6 使用了 4 个寄存器,并加进 32 比特的整数乘法,即二次函数 $B\times(2B+1)$(相应为 $D\times(2D+1)$),用于加强扩散特性.

关于 RC6 的分析结果可归纳如下:

(1) 对 RC6 的最好攻击似乎是穷举搜索用户的密钥.

(2) 对 RC6 进行差分和线性密码分析[12],所需的数据超过现有的数据.

值得注意的是,由于使用了 32 比特的整数乘法,RC6 的速度可能受到影响.

1.12 Rijndael 算法

Rijndael 算法是比利时的 Joan Daemen 和 Vincent Rijmen 提交的一个候选算法. 该算法的原形是 Square 算法,它的设计策略是宽轨迹策略(wide trail strategy)[13]. 宽轨迹策略是针对差分分析和线性分析提出的[14],它的最大优点是可以给出算法的最佳差分特征的概率以及最佳线性逼近的偏差的界;由此可以分析算法抗击差分密码分析及线性密码分析的能力.

Rijndael 采用的是代替/置换网络. 每一轮由 3 层组成,线性混合层:确保多轮之上的高度扩散;非线性层:由 16 个 S-盒并置而成,起到混淆的作用;密钥加层:子密钥简单的异或到中间状态上. S-盒选取的是有限域$GF(2^8)$中的乘法逆运算,它的差分均匀性和线性偏差都达到了最佳.

Rijndael 的安全性如下:

4 轮 Rijndael 的最佳差分特征的概率及最佳线性逼近的偏差分别为 2^{-150}和 2^{-76};8 轮 Rijndael 的最佳差分特征的概率及最佳线性逼近的偏差分别为 2^{-300}和 2^{-151};

"Square"攻击是针对 Square 算法提出的一种攻击方法,它对 Rijndael 也是适用的. 分析结果显示,7 轮以上的 Rijndael 对"Square"攻击是免疫的.

1.13 SAFER+算法

SAFER +算法是基于 SAFER 系列算法[15]提出的,因此,它的安全性可以说是经过了时间的考验. 另外,

SAFER＋算法中仅使用了字节运算，并且所需的内存小. 因此，在 Smart 卡等方面的应用是很有优势的. SAFER＋算法是一个代替/线性变换密码. 非线性层使用指数和对数运算，选择这两个函数有如下理由：(1) 使用数学中的常用函数，避免了设置陷门的可能；(2) 已证明这两个置换的分支布尔函数与随机选取的布尔函数有相同的非线性阶的项数分布. SAFER＋对以前的 SAFER 系列算法[16]的主要改进之一在于可逆线性变换 M 的选取上，它比以前的 PHT 好得多，并且 M 的选取还使得 SAFER＋可以很好地抗击差分分析.

通过对 SAFER＋的穷尽研究表明，所有的 5 轮特征概率小于 2^{-128}. 结论是 6 轮或更多轮的SAFER＋对差分密码分析是免疫的. SAFER＋继承了 SAFER 系列算法抵抗线性密码分析的能力，3 轮 SAFER＋就可抵抗线性密码分析.

不论从结构、速度，还是从安全性方面来看，SAFER＋算法都是一个很好的算法.

1.14 SERPENT 算法

SERPENT 算法是 Anderson，Biham 和 Kundsen 提交的一个候选算法. 它采用的是代替/置换网络. 在 SERPENT 的最初版本中[17]，使用了 DES 的 S-盒，目的是使公众相信设计者没有设置任何陷门. 对于 SERPENT 有类似的保证，这是因为 S-盒以简单的、确定的方式生成. SERPENT 的加密算法由 3 部分组成：第 1 部分是初始置换；第 2 部分是 32 轮的加密操作，每一轮包含密钥混合运算、S-盒及线性变换；第 3 部分是末尾置换.

SERPENT 的 S-盒是具有以下性质的 4 比特置换：

(1) 每一差分特征有一个接近 0.25 的概率，且 1 比特输入变换绝对不会导致一个 1 比特的输出变换.

(2) 每个线性逼近有一个在范围 0.5 ± 0.25 内的概率，且单独 1 比特输入与单独 1 比特输出间的线性关系有一个在范围 0.5 ± 0.125 内的概率.

(3) 每个分支布尔函数的次数为 3.

线性变换的选取原因其一是为了极大化雪崩效果，其二是为了便于分析. 设计者已证明 SERPENT 能抵抗已知的所有攻击，并且声称 SERPENT 比三重 DES 更安全. 目前还没有对 SERPENT 的进一步分析结果.

1.15 Twofish 算法

Twofish 算法是美国的 Bruce Schneier 等人提交的一个候选算法. 它的总体结构是一个 16 轮的 Feistel 结构，主要特点是 S-盒由密钥控制. Twofish 的加密分 3 部分：第 1 部分是初始白化，第 2 部分是 16 轮的加密，在第 i 轮中，令 $R_{i,0},R_{i,1},R_{i,2},R_{i,3}$ 为 4 个 32 比特的输入，4 个 32 比特的输出 $R_{i+1,0},R_{i+1,1},R_{i+1,2},R_{i+1,3}$ 用下列式子计算：

$$R_{i+1,0}=[(g(R_{i,0})+g(R_{i,1}<<<8)+K_{2i+8})\oplus R_{i,2}]>>>1,$$

$$R_{i+1,1}=(g(R_{i,0})+2g(R_{i,1}<<<8)+K_{2i+9})\oplus(R_{i,3}<<<1),$$

$$R_{i+1,2}=R_{i,0},$$

$$R_{i+1,3}=R_{i,1},$$

其中，＋是模 2^{32} 加，$<<<1$ 表示左旋转 1 比特，$>>>1$ 表示右旋转 1 比特. 第 3 部分是末尾白化.

函数 g 是 Twofish 的核心，在 g 的构作中，用到两个固定的 8×8 置换 q_0 和 q_1. q_0 和 q_1 分别利用两组 4 个 4×4 置换 t_0,t_1,t_2,t_3，用固定方法构作.

关于 Twofish 的分析结果，目前公开的最佳攻击是用 $2^{22.5}$ 个选择明文和 2^{51} 的计算量攻破 5 轮 Twofish.

2 结束语

从 AES 的征稿情况来看，目前设计分组密码仍然离不开旧的模式，无论在理论上，还是在技术上都没有什么创新. 关于 15 个 AES 候选算法的讨论在 1999 年将会很激烈，感兴趣的读者请多关注有关报道.

致谢　本文的研究得到国家自然科学基金资助，此项目编号为 69673016.

参考文献

1　AES Candidate Algorithms. http://csrc.nist.gov/encryption/aes/aes_home.htm#candidates

2 Youssef A, Chen Z, Tavares S. Construction of highly nonlinear injective S-boxes with application to CAST-like encryption algorithm. In: Proceedings of the Canadian Conference on Electrical and Computer Engineering (CCECE'97), 1997. 330～333

3 Matsui Mitsuru. Linear cryptanalysis method for DES cipher. Lecture Notes in Computer Science, Springer-Verlag, 1993,765:368～397

4 Biham E, Shamir A. Differential Cryptanalysis of Data Encryption Standard. New York: Springer-Verlag, 1993

5 Lee J, Heys H, Tavares S. Resistance of a CAST-like encryption algorithm to linear and differential cryptanalysis. Designs, Codes and Cryptography, 1997,12(3):267～282

6 Knudsen L R. Truncated and higher order differentials. Fast Software Encryption, Lecture Notes in Computer Science, Springer-Verlag, 1995,1008:196～211

7 Moriai S, Shimoyama T, Kaneko T. Higher order differential attack of a CAST cipher. Fast Software Encryption, Proceedings of 5th International Workshop. Springer-Verlag, 1998. 17～31

8 Vaudenay S. Provable security for block ciphers by decorrelation. In: STACS 98, Paris, France, Lecture Notes in Computer Science, Springer-Verlag, 1998,1373:249～275

9 Vaudenay S. The Decorrelation Technique Homepage. URL: http://www.dmi.ens.fr/vaudenay/decorrelation.html

10 Biham E, Shamir A. Differential Cryptanalysis Snefru, Kharfe, REDOC-II, LOKI and Lucifer. Lecture Notes in Computer Science, Springer-Verlag, 1991,576:156～171

11 Tokita Toshio, Sorimachi Tohru, Matsui Mitsuru. Linear cryptanalysis of LOKI and S2DES. Lecture Notes in Computer Science, Springer-Verlag, 1994,917:363～366

12 Kaliski B S, Yin Y L. On differential and linear cryptanalysis of the RC5 encrytion algorithm. Advances in Cryptology-Crypto'95. Lecture Notes in Computer Science, 1995,963:171～184

13 Daemen J, Knudsen L, Rijmen V. The block cipher square, fast software encryption. In: Proceedings of the 4th International Workshop. Springer-Verlag, 1997. 469～472

14 Daemen J. Cipher and hash function design stategies based on linear and differential cryptanalysis[Ph. D. Thesis], K. U. Leuven, 1995

15 Harpes C, Massey J. Partitioning cryptanalysis, fast software encryption. In: Proceedings of the 4th International Workshop. Springer-Verlag, 1997. 13～27

16 Massey J. SAFER K-64: a byte-oriented block-ciphering algorithm. Fast Software Encryption, Lecture Notes in Computer Science. 1994,809:1～17

17 Jakobsen T, Knudsen L. The interpolation attack on block ciphers, fast software encryption. In: Proceedings of 4th International Workshop. Springer-Verlag, 1997. 28～40

Brief Commentary on the 15 AES Candidate Algorithms Issued by NIST of USA

WU Wen-ling FENG Deng-guo QING Si-han

(*Institute of Software The Chinese Academy of Sciences Beijing* 100080)

(*Engineering Research Center for Information Security Technology The Chinese Academy of Sciences Beijing* 100080)

Abstract In this paper, the authors introduce the basic ideas of the 15 AES(advanced encryption standard) candidate algorithms issued by National Institute of Standards and Technology (NIST) of USA. The current analysis results of these algorithms are also presented.

Key words AES (advanced encryption standard), DES (data encryption standard), encryption, decryption, key.